

AD-A078 391

BURROUGHS CORP PAOLI PA FEDERAL AND SPECIAL SYSTEMS GROUP F/G 9/2
SOFTWARE MAINTENANCE MANUAL FOR THE MODULAR SYSTEM CONTROL DEVE--ETC(U)

NOV 79

DCA100-76-C-0083

UNCLASSIFIED 66157

SBIE-AD-E100 313

NL

1 OF 5
AD
A078391



ADA 078391

LEVEL

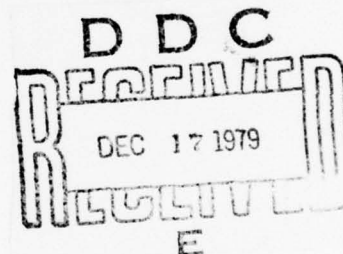
Book 1

AD-E 100 313

66157

November 1979

**SOFTWARE MAINTENANCE MANUAL
FOR THE
MODULAR SYSTEM CONTROL
DEVELOPMENT MODEL (MSCDM)**



for

**THE DEFENSE COMMUNICATIONS AGENCY
WASHINGTON, D.C. 20305**

DDC FILE COPY

Burroughs Corporation

Federal and Special Systems Group

Paoli, Pa. 19301

This document has been approved
for public release and sale; its
distribution is unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 66157	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Software Maintenance Manual for the Modular System Control Development Model (MSCDM) Book 1		5. TYPE OF REPORT & PERIOD COVERED FINAL Sep 76 - Nov 79
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s)		8. CONTRACT OR GRANT NUMBER(s) DCA100-76-C-0083
9. PERFORMING ORGANIZATION NAME AND ADDRESS Burroughs Corporation Federal and Special Systems Group Paoli, PA 19301		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS PE 33126 T&CCP 3012 - Task 15203
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Communications Agency Defense Communications Engineering Center 1860 Wiehle Ave., Reston, VA 22090		12. REPORT DATE November 1979
		13. NUMBER OF PAGES 391
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Program Listings DCS System Control Loop Network Distributed Computer System Ring Network Modular Architecture		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The Distributed Master Control Program (DMCP) and the technical control application programs of the Modular System Control Development Model (MSCDM) are described. Program listings of each program and subprogram are presented. Programs are written in FORTRAN and MACRO-11 for Digital Equipment Corporation (DEC) LSI-11 microcomputers.		

79 22 5 043

DD FORM 1473

1 JAN 73

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Book 1

18 SBIE

17

AD-E100 313

DDC
RECEIVED
DEC 17 1979
E

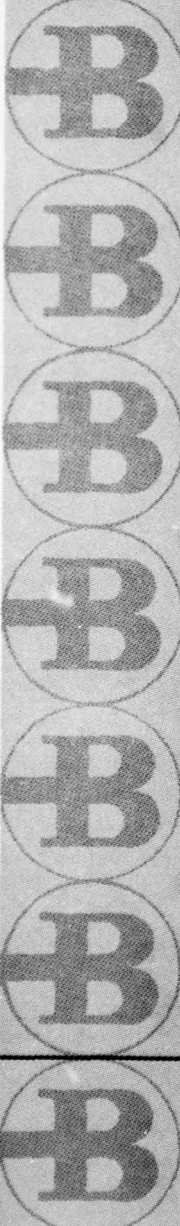
12 392

11

November 1979

14 66157

11



6
SOFTWARE MAINTENANCE MANUAL
FOR THE
MODULAR SYSTEM CONTROL
DEVELOPMENT MODEL (MSCDM)
Book 1.

9 Final rept. Sep 76 - Nov 79.

for

THE DEFENSE COMMUNICATIONS AGENCY
WASHINGTON, D.C. 20305

15 DCA100-76-C-0083

Burroughs Corporation

Federal and Special Systems Group

Paoli, Pa. 19301

070 040

This document has been approved
for public release and sale; its
distribution is unlimited.

AB

Foreword

This publication is the Software Maintenance Manual for the Modular System Control Development Model (MSCDM). This manual was prepared by the Burroughs Corporation and is submitted in accordance with the requirements of Contract DCA 100-76-C-0083.

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DDC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or special
A	

MSCDM SOFTWARE MAINTENANCE MANUAL

Book 1		PAGE
	INTRODUCTION	1
1.0	<u>MSCDM DMCP AND APPLICATION PROGRAMS</u>	3
1.1	DESCRIPTION OF DMCP	4
1.2	SSCI NODE 21	18
1.3	VSQC NODE 22	51
1.4	DSQC NODE 23	93
1.5	DBMS NODE 24	133
1.6	OCRI NODE 25	164
1.7	BWBSA NODE 26	198
1.8	FIAC NODE 27	238
1.9	SDCA NODE 28	279
1.10	USER LANGUAGE	320
1.11	SIG / SDCA - 11/40 PROGRAM	371
Book 2		
2.0	<u>UTILITY PROGRAMS</u>	
2.1	PAGE PROGRAM	
2.2	FORMAT PROGRAM	
2.3	STATUS PROGRAM	
2.4	FDMLDR PROGRAM	
3.0	<u>MACRO-11 PROGRAMS</u>	
3.1	DIAGNOSTICS LOOP 5	
3.2	IEEE PROGRAMS	
3.3	LOADER PROMS	

MSCDM SOFTWARE MAINTENANCE MANUAL

Book 2

PAGE

4.0 MODIFICATION TO ESMD SOFTWARE

4.1 GAT13 STACK MACHINE

4.2 ESMD USR LANGUAGE

4.3 ESMD LOADER

APPENDIX 1 Systems Drawings

APPENDIX 2 System Diskettes

APPENDIX 3 Startup Procedures

APPENDIX 4 Interface Data

APPENDIX 5 Acronyms Glossary

Book 3

PAGE

5.0 MSCDM Software Flow Charts

INTRODUCTION

The Modular System Control Development Model (MSCDM) consists of nine functional nodes: Station-to-Station Communications Interface (SSCI), Voice Service Quality Control (VSQC), Digital Service Quality Control (DSQC), Data Base Management Service (DBMS), Operator Control and Report Interface (OCRI), Baseband Signal Analysis and Wide Band Signal Analysis (BWBSA), Fault Isolation and Control Coordination (FIAC), Switch Data Collection and Analysis (SDCA), and Simulated Input Generator (SIG). Each of these nodes is implemented using microcomputer hardware and software, and node intercommunication is performed via a Burroughs loop architecture under control of a distributed master control program (DMCP).

Each node can communicate with any other node on the loop; however, the nodal software for the MSCDM application defines the flow of information in the system. For example, the OCRI terminal normally communicates with the DBMS node, which runs the User Language. The other ESM terminals communicate with the User Language via the loop 4-5 gateway Node 21 (SSCI).

A simulated input generator (SIG) generates inputs to the VSQC, DSQC and BWBSA, which communicate faults to the FIAC module. FIAC generates event reports to the OCRI and DBMS. The PDP 11/40 in loop 2 generates inputs to the SDCA which generates fault reports to the OCRI and DBMS. The DBMS, OCRI and FIAC communicate with the other loops via SSCI.

An LA36 DECWRITER is be used as the OCRI hard-copy terminal attached to node 25. A VT52 DECSCOPE is be used as a local CRT terminal connected to the Program Development Unit (PDU).

The Digital Equipment Corporation System's software for the PDP-11/V03 is contained on 9 - Floppy diskettes conforming to DEC's RX01 Floppy drive format.

The MSCDM applications software which runs on the PDP-11/V03 and the LSI-11/2 nodes is contained on 21- floppy diskettes.

References for the FORTRAN and MACRO-11 languages and the PDP-11/V03 system used include documentation:

1. "RT-11 System Generation Manual"
2. "Introduction to RT11"
3. "RT-11 System User's Guide"
4. "RT-11 System Message Manual"
5. "PDP-11 MACRO Language Manual"
6. "PDP-11 FORTRAN Language Manual"
7. "Advanced Programmer's Guide"

1.0 MSCDM DMCP and APPLICATION Programs

The MSCDM loop contains nine LSI-11 microprocessors, eight of which are configured with loop interface units. The ninth is a LSI-11 which contains three asynchronous interfaces, and is used as the simulated input generator to VSQC, DSQC and BWBSA. The eight loop nodes run a program called the Distributed Master Control Program (DMCP), which performs loop and exodevice interfacing and the queuing of incoming and outgoing messages. Each MSCDM loop microprocessor also runs an application program which is then linked to the DMCP to form a load module for each node. Section 1.1 provides a functional description of the DMCP and the common routines of DMCP shared by the eight loop nodes.

Sections 1.2 to 1.9 contain the description of non-common routines along with the listings of the DMCP and application programs for each node (e.g., Section 1.2 Node 21 (SSCI), Section 1.3 Node 22 (VSQC), etc.). Section 1.10 is the description and listings of the MSCDM User Language which is the application program for Node 24 (DBMS). Section 1.11 is the description and listings of the two simulated input programs used by MSCDM. At the end of each Section (1.2 - 1.11) there are listings of the command files used for compiling and linking the DMCP and application source files.

1.1 THE DISTRIBUTED MASTER CONTROL PROGRAM

The Distributed Master Control Program (DMCP) resides mainly in the microprocessor (LSI-11) nodes which make up the Burroughs Loop Architecture. DMCP modules which reside in the host computer include a Down-Line Load Module which loads remote microprocessors with memory image object files, an Initialization Module which is used at start-up time, and a System Control Resource Allocation Module which is used for dynamic system reconfiguration. In addition, the host computer stores the tables required by the node microprocessors for reconfiguration and functional task assignment.

1.1.1 DMCP Functional Description

The DMCP can be described in terms of the following functional modules: Interrupt Handlers, Protocol Modules, Loop Manager, Read and Write Modules, Queue Manager, Command Interpreter, Error Module, and I/O Queues. These functional modules are described below.

The DMCP is an Interrupt or Event Driven System. Events result in interrupts being generated by the external hardware (e.g., Loop Interface Unit - LIU) to the LSI-11 microprocessor. The interrupt results in the execution of a vectored interrupt address which passes program control to an I/O Handler (written in MACRO-11 Assembly Language).

A general understanding of the DMCP can be obtained by examining the flow of control resulting from an event. Figure 3 gives the flow control resulting from a packet being received by an LIU from the loop. The LIU generates an interrupt to its microprocessor which results in the execution of the LIU Handler. The LIU Handler examines a 16-bit LIU Status Register. Conditions which the LIU Status Register indicate include Buffer 0 or 1 Full, CRC 0 or 1 OK, Input Buffer 0 or 1 Overflow, Line Switch Primary, and Line Switch Backup. Depending on the LIU Status Register value and the information received, control can be passed to one of three modules: Error Module, Input Q Handler or Command Interpreter. The Error Module is entered if the LIU status register indicates an error condition (e.g., Line Switch Primary). An error report is generated describing the failure. The Loop Protocol Module is then entered and the Packet Header Information will be determined (e.g., sequence number destination and source process ID's). The Loop Address field is also determined for the process ID of the destination task which accepts reports. Finally the packet is written to the loop using the Write to Loop Module.

The second path in Figure 3 is followed if the input from the loop is a normal type packet destined to an Exodevice (e.g., terminal, minicomputer, data comm line). The Input Q Handler is entered to link the packet to the Input to Exodevice Queue. The Exodevice Protocol Module builds the protocol characters for the Exodevice (e.g., SOH, ETX). The Write to Exodevice Module is then called to write the packet onto the loop.

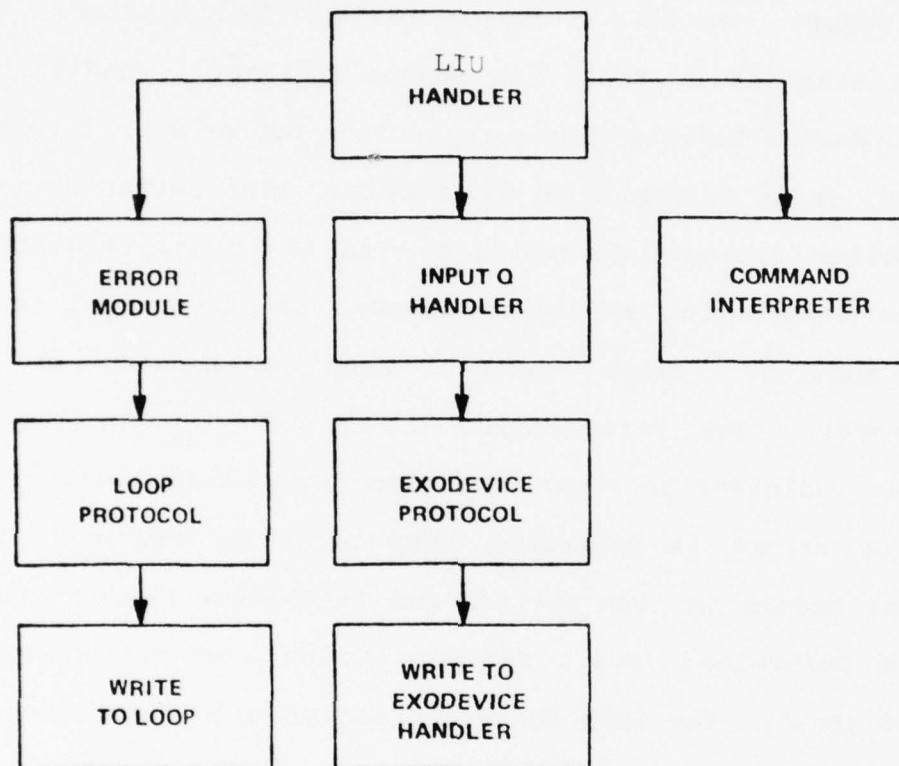


Figure 3. Input from Loop Flow Control

The third path in Figure 3 is the command interpreter function, for handling ACK and NAK messages.

The control for an input from an Exodevice is given in Figure 4. The interrupt results in the execution of the Exodevice Read Handler which can pass control to two modules. A normal loop destined packet results in the Output Q Handler being called. The Loop Protocol Module formats the header of the packet and determines the loop address. The Write to Loop Module writes the packet to the loop.

The second path in Figure 4 results in the Command Interpreter. This module interprets commands originating from the Exodevice. An example would be an ATTACH command which would be used to logically attach a peripheral connected to the loop to a minicomputer.

The DMCP Idle Flow Control is given in Figure 5. The Input Q Handler is entered to examine the current input queue size. The Output Q Handler is entered to examine whether the packet in the output queue has been ACKed or NAKed. The Loop Manager maintains the watchdog timing function for write token timeouts and retransmissions of packets that were not ACKed or NAKed. The System Monitor sends messages to its Exodevice and a System Master to monitor equipment operation.

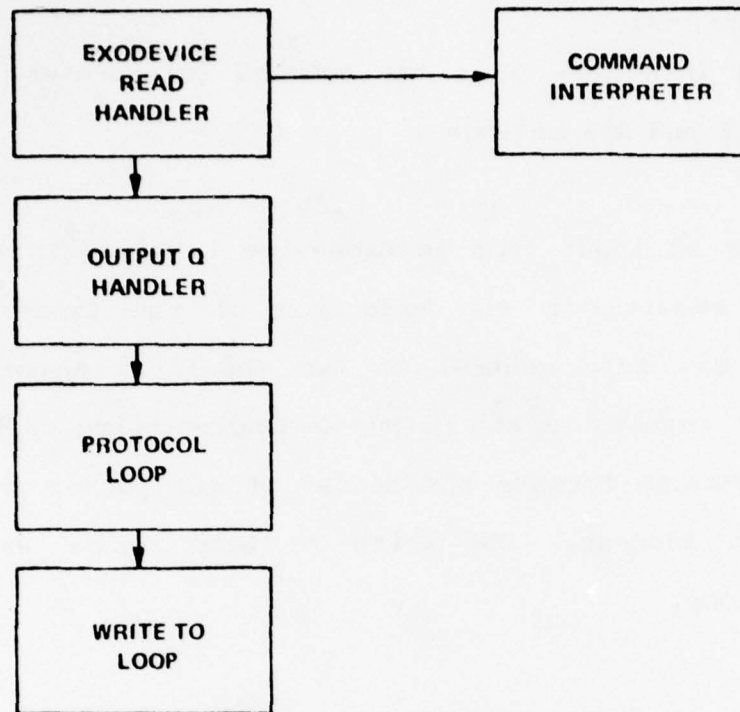


Figure 4. Input from Exodevice Flow Control

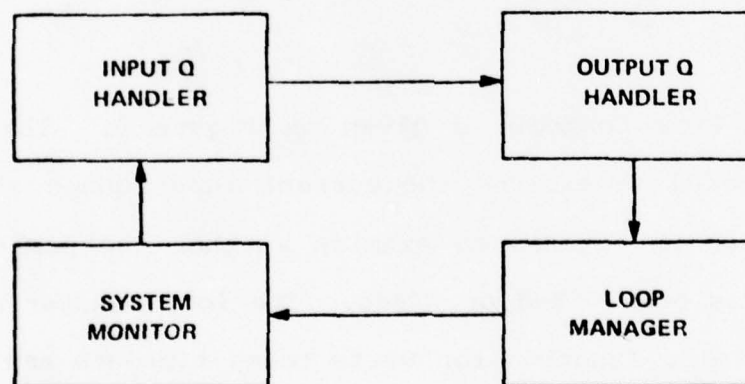


Figure 5. DMCP Idle Flow Control

1.1.1.1 Interrupt Handlers Function

Interrupt Handlers are used to process I/O interrupts. The DMCP in each node contains four basic types. The LIU Handler, examines the LIU Status Register and does a Memory Access (DMA) transfer to move a packet (at least 256 bytes) from the LIU buffers to the microprocessor memory. The Clock Handler processes interrupts from the 60 Hz Line Frequency Clock. Interrupts occur every 16.67 msec.; a clock register in memory is incremented at each interrupt. The LIU and Clock Handlers are identical for each node.

A Read and Write Exodevice is used for each type of Exodevice. Peripheral and data comm handlers receive interrupts on a per byte basis. Interrupt Handlers are written in MACRO-11 Assembly Language.

1.1.1.2 Protocol Modules Function

There are two basic types of Protocol Modules. A Loop Protocol Module and Exodevice Protocol Modules. The Loop Protocol Module is the same at all nodes. Exodevice Protocol Modules are used for each type of exodevice. Protocol Modules format packets, perform code conversion if necessary, and handle communication protocols for the loop, processors, peripherals, and data comm. The Loop Protocol Module determines destination loop addresses based upon its tables for reconfiguration and functional task assignment. Protocol Modules are either separately written in FORTRAN IV or combined with Interrupt Handlers and written in MACRO-11.

1.1.1.3 Loop Manager Function

The Loop Manager is written in FORTRAN and is identical for all nodes. It manages certain control functions associated with the loop. One such function is the regeneration of Write Tokens. A Write Token is a special circulating control packet which the LIU must recognize before it can write on the loop. If a Write Token is not received within a certain amount of time, the Loop Manager creates a new Write Token. Another function is the resending of packets to the loop that have not ACKed or NAKed within a certain amount of time.

1.1.1.4 Read Module Function

Read Modules read data from either the loop or an Exodevice. When combined with an Interrupt Handler, Read Modules are written in MACRO-11 Assembly Language.

1.1.1.5 Write Module Function

Write Modules write data to the loop or an Exodevice. When combined with an Interrupt Handler, Write Modules are written in Assembly Language.

1.1.1.6 Queue Manager Function

The Queue Manager maintains input and output queues. It maintains status on the number in queue and the current top-of-queue pointer. It moves certain control type command messages to top-of-queue or to other I/O buffers (e.g., intermediate ACK's). The Queue Manager monitors the size of queue and generates flow control command messages. The flow control messages direct nodes to stop sending to a node whose queue is nearly fully and resume sending to a node whose queue has emptied. The Queue Manager is used at all nodes. It is written in FORTRAN and interfaces to Interrupt Handlers via COMMON I/O buffers.

1.1.1.7 Command Interpreter Function

The Command Interpreter is used to perform nodal control functions as a result of interpreting command packets (utilizing the header control bytes). Many of these command packets such as ACK-NAK's, line switch, etc. are described above. A function not listed above is Down-Line Loading Command interpretation. A Down-Line Loading command forces the microprocessor to execute a bootstrap loader program stored in PROM memory. The bootstrap loader program loads data coming from the loop into the RAM memory. The data is sent to Absolute Loader Format by the loader host computer (Downline Load Module). The Loader PROM program returns program control to the first executable RAM memory location when an end-of-load command is received. The LSI-11 microprocessor is strapped such that the bootstrap loader PROM is executed at power-up time. The bootstrap loader PROM program is written in Assembly Language.

1.1.1.8 Error Module Function

The Error Module is written in FORTRAN and is used at all nodes. The Error Module formats error reports to describe nodal error conditions. These error reports include error conditions indicated by the LIU Status Register (e.g., Loopback), no response from destination nodes, and queue overflow. The reports are sent to a loop connected peripheral.

1.1.1.9 I/O Queues Function

I/O Queues are maintained by the Queue Managers. The maximum queue sizes are determined by the speed of the Exodevice Interface, the traffic at the node, and the availability memory (64K bytes maximum). Output to the loop queues is effectively of size one since each packet is ACKed before the next one is sent. This guarantees that packets are received in the same order as sent and eliminates the need for packet sequencing software.

1.1.2 DMCP ROUTINES

The DMCP Control Software comprises a set of FORTRAN and MACRO-11 routines which are linked to an application program to form a load module which is then down-line loaded to the LSI-11 nodes.

Since most of the code is common to all nodes, the description of the common routines will be divided and discussed in this section. Unique code or routines will be discussed later under the sections devoted to each node (i.e., Sections 1.2 to 1.9).

1.1.2.1 Program Nodal Description (FORTRAN)

This is the main program of the DMCP and its functions are to examine the queues to see if they are empty or full, move data to empty queues, perform output to the LIU and exodevices, and check for timeouts, for write token detect, and LIU output message acknowledgement. It also calls the application program when there is something in the loop input queue.

1.1.2.2 Subroutine IGETSP (FORTRAN)

The purpose of this subroutine is to fetch a pointer to a free packet array from the free packet list. The free packet list is examined to determine if the list is empty; if it is, the INIT subroutine is called. If not, a pointer to an available packet is fetched.

1.1.2.3 Subroutine ENQUE (FORTRAN)

This subroutine adds an element at the tail of a queue. If the current tail is not at the lower limit of the queue, the new element is appended and the tail pointer adjusted. Otherwise, a

check is made to determine whether or not the ENQUE is full. If the queue is not full, the queue is moved so that the head is at the upper limit and the new element is added. If full, subroutine INIT is called.

1.1.2.4 Subroutine DEQUE (FORTRAN)

This subroutine returns the head of a specified queue element. The element may or may not be removed from the queue, depending upon a parameter setting.

1.1.2.5 Subroutine ACKNAK (FORTRAN)

This subroutine is called when an ACK or NAK message is received. In the case of the ACK message, the output queue is marked empty, the pointer to the packet pointed to by the output queue is returned to the free packet list and the write flags are reset.

1.1.2.6 Subroutine INPTQ (FORTRAN)

This subroutine checks the validity of a non-control message received from the LIU. If the message is valid, an ACK message is generated and placed in the ACK queue. If the message is not valid, a NAK message is generated.

1.1.2.7 Subroutine LPINPT (FORTRAN)

This procedure determines whether a message received from the LIU is a control message, an ACK or NAK message, or a normal message. The appropriate subroutine is called after the determination has been made. For ACK's and NAK's, subroutine ACKNAK is called. For normal messages, subroutine INPTQ is called for normal messages.

1.1.2.8 Subroutine LINLOS (FORTRAN)

This subroutine assembles a message indicating the switching of the primary or backup lines. The message is sent to Node 25 and printed on the LA-36.

1.1.2.9 Subroutine INIT (FORTRAN)

This subroutine initializes the queues, the free arrays, the LID of the node, the read address of the node, the LID/FAD table and the address comparison RAM of the LIU.

1.1.2.10 Subroutine MASTER (MACRO)

This subroutine initializes all interrupt vectors of the devices of the node, lowers the CPU priority to allow interrupts to be processed.

1.1.2.11 Subroutine LIUINT (MACRO)

This subroutine initializes the LIU's, address comparison RAM, both input and output buffers, and the status register.

1.1.2.12 Subroutine LIO (MACRO)

This subroutine is entered by an interrupt from the LIU. It reads the status of the LIU and determines what caused the interrupt. If one of the input buffers is full, it unloads them and calls subroutine LPINPT. If it was a line loss, it calls subroutine LINLOS, and if it was a bad CRC on either of the input buffers, it unloads the buffers and calls subroutine LPINPT.

1.1.2.13 Subroutine ENABLE (MACRO)

This subroutine is passed a variable which either lowers the CPU priority to enable interrupts or raises priority thus disabling interrupts from being acknowledged.

1.1.2.14 Subroutine Switch (MACRO)

This subroutine is passed a variable which either sets or resets the backup or primary line switches.

1.1.2.15 Subroutine Status (MACRO)

This subroutine is passed one argument that specifies which LIU status byte is to be read. That status byte is then passed back as the second argument in the call statement.

1.1.2.16 Subroutine WTOKEN (MACRO)

This subroutine is called to regenerate a write token in the loop.

1.1.2.17 Subroutine TIME (MACRO)

The subroutine is the interrupt routine that is called by the LTC interrupt. It increments a variable and checks for counter overflow.

1.1.2.18 Subroutine RAM (MACRO)

The subroutine is called to read or write a location in the LIU address comparison RAM.

1.1.2.19 Subroutine LPOUT (MACRO)

This subroutine is called to load output buffer 0 with a message, and load output buffer 1 with a write token.

1.1.2.20 Subroutine RDPNT (MACRO)

This subroutine is called to read a buffer pointer of either input buffer 0 or 1 or output buffer 0 or 1.

1.1.2.21 Subroutine RSTART (MACRO)

The subroutine is called to either restart nodal software, halt processor or jump to loader prom.

1.2 NODE 21 (SSCI)

The Station-to-Station Communication Interface (SSCI) serves as a gateway node interface to loop 4 of the ESM. The loop 4 - loop 5 interface implemented is 9600 baud asynchronous. The SSCI is used to simulate communication between different system control sites. The SSCI performs code conversion, intransit queueing and packet routing.

1.2.1 Program Description

1.2.1.1 Refer to Section 1.1 for descriptions of routines - IGETSP, ENQUE, DEQUE, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME, RAM, LPOUT, RDPNT, RSTART.

1.2.1.2 Program NODAL Description (FORTRAN)

This is the main program for the SSCI DMCP and its functions are to examine the queues to see if they are empty or full, move data to empty queues, write the output of all the incoming loop messages to the gateway interface (Node 13, loop 4) and place on loop 5 all messages received from the gateway interface, SSCI does not acknowledge messages received from the loop, ESM employs "end to end" acknowledgement (SSCI functions only as the gateway).

1.2.1.3 Subroutine INPTQ (FORTRAN)

This subroutine does the queueing of incoming packet from the way. No check is made for control packets.

1.2.1.4 Subroutine LPINPT (FORTRAN)

This subroutine only calls INPTQ routine because SSCI does not scan any of the incoming messages from the loop interface. All incoming messages go to the gateway interface, also LPINPT do not generate ACK or NAK to messages.

1.2.1.5 Subroutine GIO (MACRO)

This subroutine is the interrupt handler for the gate-way interface. It reads the message from the interface and then sets a flag containing the byte count of that message.

1.2.1.6 Subroutine GOUT (MACRO)

This subroutine is called to send a message over the gateway interface to node 13, loop 4. It's passed one argument the byte count of the message to be sent.

PAGE 001

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:07:01

```

0001      PROGRAM NODAL
0002      INTEGER*2 XINQ,XOUTQ,ACKQ,FXINQ,FXOUTQ,PACKQ
0003      INTEGER*2 PINQ,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP,CC1,CC2
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFQ,OUTBF,INBF
0007      LOGICAL*1 PACK,ETX,CR,LF,MONITOR,ISLID,DUM
0008      INTEGER*2 LTIME,ACKTIM,NEWTIM,OLDTIM,IWRITM,TIMLIM,ATIMLM
0009      COMMON /MESS/ MESSEQ
0010      COMMON /DFM/ OUTBF(256),INBF(256),IWRITM,IWRIT,STAT,LTIME
0011      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012      1      ACKQ(16),PACKQ(2),INQ(16),FINQ(2)
0013      1      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0014      1      LIDFQ(256),IOFLG,LLFLG,IRSEND
0015      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0016      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0017      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
0018      1      RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0019      CALL MASTER
0020      CALL INIT
0021      CALL ENABLE(1)
0022      5      CONTINUE
0023      25      IF(PINQ(1).LT. PINQ(2)) GOTO 40
0024      CALL ENABLE(0)
0025      Q2=DEQUE(PINQ,INQ,1)
0026      CALL ENABLE(1)
0027      CALL DESTQ(PACK(254,Q2),LEN)
0028      DO 30 I=1,LEN
0029      XOUTBF(I)=PACK(I,Q2)
0030      30      CONTINUE
0031      CALL GOUT(LEN,IST)
0032      IFR=IFR+1
0033      FREE(IFR)=Q2
0034      40      CALL STATBQ(15)
0035      IF(15.EQ. 1) GOTO 110
0036      IF(IOFLG.EQ. 0) GOTO 81
0037      83      CALL ENABLE(0)
0038      Q2=IGETSP(N)
0039      CALL ENABLE(1)
0040      21      DO 82 I=1,IOFLG
0041      PACK(I,Q2)=XINBF(I)
0042      82      CONTINUE
0043      CALL ENSTR(PACK(254,Q2),IOFLG)
0044      CALL ENABLE(0)
0045      CALL ENQUE(PXINQ,XINQ,Q2)
0046      CALL ENABLE(1)
0047      IOFLG=0
0048      84      IF(110.EQ. 0) CALL LINLOS
0049      81      IF(PXINQ(1).LT. PXINQ(2)) GOTO 110
0050      CALL ENABLE(0)
0051      Q1=DEQUE(PXINQ,XINQ,1)
0052      CALL ENABLE(1)
0053
0054
0055
0056

```

PAGE 002

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:07:01

```

0057      OUTQ=Q1
0058      85      CALL DESTR(PACK(254,OUTQ),Q2)
0059              DO 90 I=1,Q2
0060                  OUTRF(I)=PACK(I,OUTQ)
0061              90      CONTINUE
0062                  OUTBF(Q2+1)=0
0063                  IPT=OUTBF(5)
0064                  OUTBF(Q2+2)=LIDFD(IPT)
0065                  CALL LPOUT(Q2+2)
0066                  IWRTH=0
0067                  IWR=1
0068              110      OLDTIM=NEWTIM
0069                  NEWTIM=LTIME
0070                  IF(IWR.EQ. 0) GOTO 130
0072                  IWRTH=IWRTH+(NEWTIM-OLDTIM)
0073                  IF(IWRTH.LT. TIMLIM) GOTO 130
0075                  CALL WTOKEN
0076                  IWRTH=0
0077                  IWR=0
0078              130      CONTINUE
0079                  GOTO 5
0080                  END

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:07:20      PAGE 001

0001      FUNCTION IGETSP(N)
0002      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003      INTEGER*2 FREE,FLWCNT
0004      INTEGER*2 OUTFCT,OUTQ,RESNLM
0005      COMMON /FRE/ FREE(64),IFR,IFRSZ
0006      1      RESNLM,IALTRI,ISLID,MONTOR,LF,ETX,CR,DUM
0007      IF(IFR.LT. 1) CALL INIT
0009      IGETSP=FREE(IFR)
0010      IFR=IFR-1
0011      RETURN
0012      END

```



```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:07:31      PAGE 001

0001      SUBROUTINE ENQUE(A,B,N)
0002      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004      INTEGER*2 FLWCNT,OUTQ,A(2),B(16)
0005      INTEGER*2 RESNLM,OUTFCT
0006      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007      1      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0008      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1      RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0010      10      IQHEAD=A(1)
0011      IQTAIL=A(2)
0012      IF(IQTAIL.EQ.1) GOTO 20
0013      IQTAIL=IQTAIL-1
0014      B(IQTAIL)=N
0015      A(2)=IQTAIL
0016      GOTO 999
0017      20      IF(IQHEAD.GE.IQLNTH) GOTO 40
0018      NN=IQHEAD-IQTAIL
0019      DO 30 I=1,NN+1
0020      30      B(IQLNTH+1-I)=B(IQHEAD+1-I)
0021      A(1)=IQLNTH
0022      A(2)=IQLNTH-NN
0023      GOTO 10
0024      40      CALL INIT
0025      999      RETURN
0026      END

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:07:43      PAGE 001

0001      FUNCTION DEQUE(A,B,ID)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOK
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,FXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004      INTEGER*2 FLWCNT,A(2),B(16),DEQUE,OUTFCT,OUTQ,RESNLM
0005      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      1      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0007      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1      RESNLM,IALTRT,ISLID,MONTOK,LF,ETX,CR,DUM
0009      IQHEAD=A(1)
0010      DEQUE=B(IQHEAD)
0011      IF(ID.NE. 1) GOTO 999
0012      IF(IQHEAD.NE. 0) GOTO 10
0013      A(1)=IQLNTH
0014      A(2)=IQLNTH+1
0015      GOTO 999
0016      10 A(1)=IQHEAD-1
0017      999 RETURN
0018      END

```

```

FORTAN IV      V02.1-1      Mon 04-Jun-79 17:08:05      PAGE 001

0001      SUBROUTINE INPTQ(L)
0002      INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ
0003      INTEGER*2 PACKQ,PINQ,FLWNT,OUTFCT,T1,RESNLM,OUTQ
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0005      LOGICAL*1 PACK,MONTOR,ISLID,ETX,CR,LF,DUM
0006      INTEGER*2 LTIME,IWRTIM
0007      COMMON /DFM/ OUTBF(256),INBF(256),
0008      1      IWRITM,IWRT,STAT,LTIME
0009      1      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010      1      LIDFD(256),IOFLG,LLFLG,IRSEND
0011      1      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012      1      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0013      1      COMMON /GLOB/ ISENT,FLWNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0014      1      RESNLM,IALRT,ISLID,MONTOR,LF,ETX,CR,DUM
0015      IF(L.LT. 1) GOTO 999
0016      LI=L-2
0017      T1=IGETSP(N)
0018      DO 10 I=1,LI
0019      10      PACK(I,T1)=INBF(I)
0020      CALL ENSTR(PACK(254,T1),LI)
0021      CALL ENQUE(PINQ,INQ,T1)
0022      RETURN
0023      999      END

```

PAGE 001

Mon 04-Jun-79 17:08:19

V02.1-1

FORTRAN IV

```

0001 SUBROUTINE LPINPT(LI)
0002 INTEGER*2 STAT,FLWCNT,OUTFCT,OUTQ,RESNLM
0003 INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004 LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONITOR,ISLID,DUM
0005 INTEGER*2 LTIME,IWRTTH
0006 COMMON/DEM/OUTBF(256),INBF(256),IWRTTH,IWRT,STAT,LTIME
0007 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
      1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0008 40 CALL INPTQ(LI)
0009 RETURN
0010 END

```


PAGE 001

Mon 04-Jun-79 17:08:30

FORTRAN IV

V02.1-1

```

0001 SUBROUTINE INIT
0002 REAL*4 RH,RL,AH,AL,IRSV,UMEAS
0003 REAL*8 UTR
0004 INTEGER*2 STAT,FLWCNT,OUTFCT
0005 INTEGER*2 SETPRM,RSTPRM,SETBNP,OUTQ,RESNLM,RSTBNP
0006 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0007 LOGICAL*1 PACK,ETX,CR,LF,MONITOR,ISLID,DUM,DATA
0008 INTEGER*2 LTIME,OLDTIM,TIMLIM,ACNTIM,ATIMLM,IWRTTH
0009 COMMON /DFM/ OUTBF(256),INBF(256),
0010 1 IWRTTH,IWRT,STAT,LTIME
0011 1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012 1 COMMON /MESS/ MESSQ
0013 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSER(256),LIDFD(256),
0014 1 IOFLG,LLFLG,IRSEND
0015 COMMON /FRE/ FREE(64),IFR,IFRSZ
0016 COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0017 COMMON /SWT/ SETPRM,RSTPRM,SETBNP,RSTBNP
0018 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0019 1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0020 CALL RAM(0,4,4)
0021 CALL RAM(0,1,4)
0022 CALL RAM(0,255,0)
0023 MESSQ=0
0024 IOFLG=0
0025 IRSEND=0
0026 LLFLG=0
0027 IFRSZ=64
0028 INFLCT=0
0029 ISLID=21
0030 CR=.015
0031 LF=.012
0032 ETX=.003
0033 RESNLM=2
0034 IQLNTH=16
0035 IALTRT=0
0036 FLWCNT=0
0037 IWRTTH=0
0038 TIMLIM=35
0039 SETPRM=4
0040 RSTPRM=32
0041 SETBNP=8
0042 RSTBNP=64
0043 ATIMLM=200
0044 MONITOR=25
0045 IWRT=0
0046 OUTFCT=0
0047 IFULL=0
0048 DO 40 I=1,20
0049 LIDFD(I)=4
0050 LIDFD(21)=1

```

PAGE 002

Mon 04-Jun-79 17:08:30

V02.1-1

FORTRAN IV

```

0051 LIDFD(32)=3
0052 LIDFD(23)=6
0053 LIDFD(24)=5
0054 LIDFD(25)=7
0055 LIDFD(26)=8
0056 LIDFD(27)=9
0057 LIDFD(28)=2
0058 DO 50 I=29,39
0059 LIDFD(I)=0
0060 DO 60 I=40,44
0061 LIDFD(I)=1
0062 DO 70 I=45,59
0063 LIDFD(I)=0
0064 DO 80 I=60,64
0065 LIDFD(I)=2
0066 DO 90 I=65,79
0067 LIDFD(I)=0
0068 DO 100 I=80,84
0069 LIDFD(I)=4
0070 DO 110 I=85,99
0071 LIDFD(I)=0
0072 DO 120 I=100,104
0073 LIDFD(I)=5
0074 DO 130 I=105,256
0075 LIDFD(I)=0
0076 PING(1)=IQLNTH
0077 PING(2)=IQLNTH+1
0078 PACKQ(1)=IQLNTH
0079 PACKQ(2)=IQLNTH+1
0080 PXOUTQ(1)=IQLNTH
0081 PXOUTQ(2)=IQLNTH+1
0082 PXINQ(1)=IQLNTH
0083 PXINQ(2)=IQLNTH+1
0084 DO 10 I=1,IFRSZ
0085 PACK(255,I)=0
0086 DO 20 I=1,IFRSZ
0087 DO 20 I=1,IFRSZ
0088 FREE(I)=I
0089 DO 30 I=1,256
0090 ACKSEQ(I)=256
0091 IFR=IFRSZ
0092 RETURN
0093 END

```

PAGE 001

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:08:50

```

0001 SUBROUTINE LINLOS
0002 REAL*8 RM1(5),LIN08,LIN18
0003 INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM,T1
0004 INTEGER*2 XING,PXING,XOUTQ,ACKQ,PACKQ,PINQ
0005 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0006 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0007 LOGICAL*1 PACK,M1(40),LIN0(8),LIN1(8)
0008 COMMON/QUE/XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
      * ACKQ(16),PACKQ(2),PING(16),PINQ(2)
0009 COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
      * IOFLG,LLFLG,IRSEND
0010 COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
      * RESNLM,IALRT,ISLID,MONTOR,LF,ETX,CR,DUM
0011 COMMON PACK(256,64)
0012 COMMON /MESS/ MESSEQ
0013 DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0014 DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0015 DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0016 DATA RM1(5)/'DE 21 '//
0017 EQUIVALENCE(LIN0,LIN08)
0018 EQUIVALENCE(LIN1,LIN18)
0019 EQUIVALENCE(M1,RM1)
0020 IS=LLFLG
0021 CALL ENABLE(0)
0022 T1=IGETSP(N)
0023 CALL ENABLE(1)
0024 PACK(1,T1)=0
0025 IF(MESSEQ.EQ.126) MESSEQ=0
0026 MESSEQ=MESSEQ+1
0027 PACK(2,T1)=MESSEQ
0028 PACK(3,T1)=0
0029 PACK(4,T1)=0
0030 PACK(5,T1)=25
0031 PACK(6,T1)=ISLID
0032 DO 20 I=7,9
0033   PACK(I,T1)=LF
0034   PACK(I,T1)=LF
0035 CONTINUE
0036 DO 30 I=1,22
0037   PACK(I+9,T1)=M1(I)
0038   IF(I5.EQ.1) GOTO 50
0039   DO 40 I=1,8
0040     PACK(I+31,T1)=LIN0(I)
0041     GOTO 70
0042   DO 60 I=1,8
0043     PACK(I+31,T1)=LIN1(I)
0044     DO 80 I=23,40
0045       PACK(I+17,T1)=M1(I)
0046       PACK(58,T1)=CR
0047       PACK(59,T1)=LF
0048       PACK(60,T1)=ETX
0049       CALL ENSTR(PACK(254,T1),60)
0050       CALL ENABLE(0)
0051       CALL ENQUE(PXING,XING,T1)
0052       CALL ENABLE(1)
0053

```

PAGE 002

Mon 04-Jun-79 17:08:50

FORTRAN IV V02.1-1

0054 LLFLG=0
0055 RETURN
0056 END


```

1  .TITLE FDM.MACRO
2  .SRITL NODE21
3  .IDENT /V3.0/
4  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WTOKEN,RAM,STATUS,STAI40
6  .GLOBL LFOUT,RSTART,RDPNT,DESTR,ENSTR,LFINPT
7  .NLIST CND
8  .PSECT
9
10
11      000000
12      000001
13      000002
14      000003
15      000004
16      000005
17      000006
18      000007
19
20
21
22      172410
23      172412
24      172414
25      172416
26      172416
27      177560
28      177562
29      177564
30      177564
31      177566
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

```

```

      R0      =%0
      R1      =%1
      R2      =%2
      R3      =%3
      R4      =%4
      R5      =%5
      SP      =%6
      PC      =%7

;INTERFACE ADDRESS
      BAR= 172410
      WCR= 172412
      CSR= 172414
      IOBUF= 172416
      OFREG= 172416
      HRCR= 177560
      HRBUF= 177562
      HXCSR= 177564
      HXRUF= 177566

;INTERFACE OPTIONS
      XCIO=1
      XSIO=1
      XGIO=0
      NDX=0
      ND24=1
      NHOST=0
      DHOST=1

;LOCAL VARIABLES
      ARG1= 2
      ARG2= 4
      ARG3= 6
      ARG4= 10

      BIT15= 100000
      BIT14= 40000
      BIT08= 400
      BIT07= 200
      BIT06= 100
      BIT05= 40
      BIT04= 20
      BIT03= 10
      BIT02= 4

```

FDM-MACRO
NODE21

MACRO V03.01 4-JUN-79 17:09:17 PAGE 1-1

66
67
68

000002
000001

BIT01= 2
BIT00= 1

```

*****
**** PROCEDURE MASTER START UP ****
*****

MASTER: MOV    $340, R0
          MTPS   R0
          MOV    $124, R0
          MOV    $110, (R0)+
          MOV    $340, (R0)
          RESET
          MOV    $60, R0
          MOV    $GID, (R0)+
          MOV    $340, (R0)+
          BIS    BIT06, @HRCSCR
          MOV    $100, R0
          MOV    $TIME, (R0)
          BIC    $A0100, @CSR
          JSR    PC, LIUINT
          CLR    @CSR
          MOV    $000, R0
          MTPS   R0
          RTS    PC

;*****
;**** PROCEDURE INITIALIZE ****
;*****

;CLEAR ACRAM
LIUINT: MOV    $4354., @OPREG
          MOV    $2304., @OPREG
          TSTB   @CSR
          BFL    -4
          CLRB   @CSR
          MOV    $4353., @OPREG
          MOV    #-256., R0
          MOV    $2311., @OPREG
          TSTB   @CSR
          BFL    -4
          CLRB   @CSR
          INC     R0
          BNE    1$

          ;CLEAR INPUT/OUTPUT BUFFERS
          MOV    $10410, R1
          MOV    $10440, R0
          JSR    ZERORP
          MOV    $1400, @OPREG
          TSTB   @CSR
          BFL    -4
          CLRB   @CSR
          MOV    $10610, R1
          MOV    $10640, R0

          ;DRUFADR CMD
          SEL INBUF0
          SET POINTER=0
          FALSE READ DATA
          GOOD READ
          NO RETRY
          CLEAR DONE BIT
          DRUFADR COMMAND
          SEL INBUF1 COMMAND

```

```

98 000234 004767 000052 JSR      PC,      ZEROEP
99 000240 012737 001400 MOV      #1400, 0#OPREG
100 000246 105737 172416 TSTB    @#CSR
101 000252 100375      BPL      -4
102 000254 105037 172414 CLRB    @#CSR
103 000260 012701 010510 MOV      #10510, R1
104 000264 012700 010540 MOV      #10540, R0
105 000270 004767 000016 JSR      PC,      ZEROEP
106 000274 012701 010710 MOV      #10710, R1
107 000300 012700 010740 MOV      #10740, R0
108 000304 004767 000002 JSR      PC,      ZEROEP
109 000310 000440      BR      STAT$
110 000312 010137 172416 ZEROEP: MOV      R1,      0#OPREG
111 000316 012737 001400 MOV      #1400, 0#OPREG
112 000324 105737 172414 TSTB    @#CSR
113 000330 100375      BPL      -4
114 000332 105037 172414 CLRB    @#CSR
115 000336 013702 172416 MOV      @#OPREG, R2
116 000342 042702 177400 BIC      #177400, R2
117 000346 010037 172416 MOV      R0,      0#OPREG
118 000352 032702 000000 CMP      #0,      R2
119 000356 001412      BEQ      3$
120 000360 012737 001400 MOV      #1400, 0#OPREG
121 000366 105737 172414 TSTB    @#CSR
122 000372 100375      BPL      -4
123 000374 105037 172414 CLRB    @#CSR
124 000400 005302      DEC      R2
125 000402 000763      BR      2$
126 000404 010037 172416 MOV      R0,      0#OPREG
127 000410 000207      RTS      PC
128
129
130
131 000412 012737 010400 STAT$: MOV      #4352, 0#OPREG
132 000420 012737 002400 MOV      #1280, 0#OPREG
133 000426 012737 002400 MOV      #1280, 0#OPREG
134 000434 000207      RTS      PC
135
136

;POINTER=0
;FALSE READ DATA
;GOOD RD
;NO RETRY
;CLEAR DONE BIT
;RDBUFADR COMMAND
;SEL OUTBUFO COMMAND

;RDBUFADR COMMAND
;SEL OUTBUFO COMMAND
;ZERO BUFFER POINTER
;GO CLEAR STATUS
;RDBUFADR
;RD
;GOOD RD
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;FETCH POINTER
;CLEAR MST BYTE
;SEL BUFFER
;POINTER=0?
;YES RECHECK
;NO FALSE RD
;GOOD RD
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;LOCAL POINTER-1
;BR UNTIL 0
;SEL BUFFER
;RETURN

;WCR:RS
;RS(FALSE)
;RS

```


[illegible]

58	000730	132767	000002	000254	R1\$:	R1B	#BIT01, CAUSE	#INBU1 FULL
59	000736	001423				REQ	RTI\$	#NO WERE DONE
60	000740	012701	000400			MOV	#INBF, R1	#BUFFER ADDRESS
61	000744	012702	010610			MOV	#4488., R2	#RDBUFAUR COMM
62	000750	012703	010640			MOV	#4512., R3	#SEL INBU1
63	000754	012704	000002			MOV	#BIT01, R4	#CRC BIT
64	000760	004767	000046			JSR	PC, EMBF	#GO EMPTY BUFFER
65	000764	012705	001214			MOV	#AREA, R5	#DATA LINK AREA
66	000770	012767	000001	000216		MOV	#1, AREA	#ONE VARIABLE
67	000776	010267	000216			MOV	R2, DATA	#CRC OR BYTE COUNT
68	001002	004767	000000G			JSR	PC, LFINPT	#CALL FORTRAN QUE'ER
69								
70	001006	012605			RTI\$:	MOV	(SP)+, R5	#RESTORE REGISTERS
71	001010	012604				MOV	(SP)+, R4	
72	001012	012603				MOV	(SP)+, R3	
73	001014	012602				MOV	(SP)+, R2	
74	001016	012601				MOV	(SP)+, R1	
75	001020	012600				MOV	(SP)+, R0	
76	001022	052737	040000	172414		BIS	#BIT14, @#CSR	#ENABLE INTERRUPTS
77	001030	000002				RTI		#RETURN FROM INTERRUPT
78								
79	001032	012737	010600	172416	EMBF:	MOV	#4480., @#DFREG	#READ STATUS 1
80	001040	012737	002400	172416		MOV	#1280., @#DFREG	#RS
81	001046	013700	172416			MOV	@#IOBUF, R0	#FETCH STATUS
82	001052	130400				BITB	R4, R0	#GOOD CRC
83	001054	001002				BNE	CRCOK	
84	001056	012704	177777			MOV	#-1, R4	#NO FLAG
85	001062	010237	172416		CRCOK:	MOV	R2, @#DFREG	#RDBUFAUR
86	001066	012737	001400	172416		MOV	#768., @#DFREG	#RD
87	001074	105737	172414			TSTB	@#CSR	#GOOD RD
88	001100	100375				BPL	-4	
89	001102	013702	172416			MOV	@#IOBUF, R2	#REPLACE WITH POINTER
90	001106	042702	177400			BIC	#177400, R2	#CLEAR MST BITS
91	001112	010200				MOV	R2, R0	#SAVE IT
92	001114	005400				NEG	R0	#2'S COMP
93	001116	010037	172412			MOV	R0, @#WCR	#BYTE COUNT NOW
94	001122	010137	172410			MOV	R1, @#BAR	#ADDRESS IN MEMORY
95	001126	010337	172416			MOV	R3, @#DFREG	#SEL BUFFER
96	001132	012737	001400	172416		MOV	#768., @#DFREG	#FALSE RD
97	001140	105737	172414			TSTB	@#CSR	#DONE ON
98	001144	100375				BPL	-4	#NO LOOP
99	001146	012737	021000	172416		MOV	#8704., @#DFREG	#FIRE DMA
100	001154	000240				NOP		#DELAY
101	001156	105737	172414			TSTB	@#CSR	#GOOD DMA
102	001162	100401				BMI	DMAOK	
103	001164	000240				NOP		
104	001166	012737	004400	172416	DMAOK:	MOV	#2304., @#DFREG	#ERROR IF HERE
105	001174	105737	172414			TSTB	@#CSR	#FALSE WD
106	001200	100375				BPL	-4	#GOOD WD
107	001202	005704				TST	R4	#WAS CRC OK
108	001204	100001				BPL	END\$	#YES
109	001206	010402				MOV	R4, R2	#NO FLAG IT
110	001210	000207				MOV	PC	#RETURN
111	001212	000000			END\$:	RTS	0	#STATUS BYTE 0 HOLDER
112	001214	000000			CAUSE:	.WORD	0	
113	001216	001220			AREA:	.WORD	0	
114	001220	000000			DATA:	.WORD	0	


```
CALL STATUS(X,DATA) -READ STATUS BYTES 0/1
;X=0 STATUS BYTE 0
;X=1 STATUS BYTE 1
```

[illegible]


```

*****
**** PROCEDURE ACRAM
*****
;CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;0100 =DREAD
;0111 =NULL
;0000 =WOKEN

```

14	001704	017500	000002	MOV	RAM:	0ARG1(R5), R0	; RAM
15	001710	022700	000000	CMP		R0, R0	; WHICH OPERATION
16	001714	001035		BNE		RDRAM	; READ OP
17	001716	017500	000004	MOV	WTRAM:	0ARG2(R5), R0	; ADDRESS
18	001722	012501	000006	MOV		0ARG3(R5), R1	; WRITE DATA
19	001725	012737	010402	MOV		0354., 00PREG	;SEL LDACR
20	001734	062700	004400	ADD		0304., R0	;WD/DATA (ADDR)
21	001740	010037	172416	MOV		R0, 00PREG	;WRITE DATA
22	001744	105737	172414	TSTB		00CSR	;VALID WRITE
23	001750	100375		BPL		-4	;NO LOOP UNTIL READY
24	001752	105037	172414	CLRB		00CSR	;CLEAR DONE BIT
25	001756	012737	010401	MOV		0353., 00PREG	;SEL ACRAM
26	001764	062701	004400	ADD		0304., R1	;WD/DATA (CMD)
27	001770	010137	172416	MOV		R1, 00PREG	;WRITE
28	001774	105737	172414	TSTB		00CSR	;VALID WRITE
29	002000	100375		BPL		-4	;NO LOOP UNTIL READY
30	002002	105037	172414	CLRB		00CSR	;CLEAR DONE BIT
31	002005	000307		RTS		PC	
32	002010	017500		MOV	RDRAM:	0ARG2(R5), R0	; ADDRESS TO READ
33	002014	012737	010402	MOV		0354., 00PREG	;SEL LDACR
34	002022	062700	004400	ADD		0304., R0	;WD/DATA (ADDR)
35	002026	010037	172416	MOV		R0, 00PREG	;WRITE DATA
36	002032	105737	172414	TSTB		00CSR	;VALID WRITE
37	002036	100375		BPL		-4	;NO LOOP UNTIL READY
38	002040	105037	172414	CLRB		00CSR	;CLEAR DONE BIT
39	002044	012737	010401	MOV		0353., 00PREG	;SEL ACRAM
40	002052	012737	001400	MOV		0768., 00PREG	;READ DATA
41	002060	105737	172414	TSTB		00CSR	;VALID READ
42	002064	100375		BPL		-4	;NO LOOP UNTIL READY
43	002068	105037	172414	CLRB		00CSR	;CLEAR DONE BIT
44	002072	013700	172416	MOV		00ORUF, R0	;FETCH DATA
45	002076	042700	177760	BIC		017760, R0	;CLEAR BITS
46	002102	010075	000006	MOV		R0, 0ARG3(R5)	;RTN DATA
47	002106	000207		RTS		PC	
48							
49							

```
1 1
2 2
3 3
4 4
5 5
6 002110 017502 000002 172410 000000' 172410
7 002114 012700 000340
8 002120 106400
9 002122 012737 000000' 172410
10 002130 005402
11 002132 010237 172412
12 002136 012737 010540 172416
13 002144 012737 024000 172416
14 002152 000240
15 002154 105737 172414
16 002160 000240
17 002162 105037 172414
18 002166 012737 010740 172416
19 002174 012737 001660' 172410
20 002202 012702 000002
21 002206 005402
22 002210 010237 172412
23 002214 012737 024000 172416
24 002222 000240
25 002224 105737 172414
26 002230 000240
27 002232 105037 172414
28 002236 012737 010420 172416
29 002244 012737 004403 172416
30 002252 105737 172414
31 002256 100375
32 002260 105037 172414
33 002264 012700 000000
34 002270 106400
35 002272 000207
36
37
38
```

***** PROCEDURE WRITE LOOP (LPOUT) *****

ENABLE LSB

LPOUT: MOV @ARG1(R5), R2
MOV #340, R0
MTFS R0
MOV @OUTBF, @BAR
NEG R2
MOV R2, @WCR
MOV #4448, @OPREG
MOV #10240, @OPREG
NOP
TSTB @CSR
NOP
CLRB @CSR
MOV #4576, @OPREG
MOV @WID, @BAR
MOV #2, R2
NEG R2
MOV R2, @WCR
MOV #10240, @OPREG
NOP
TSTB @CSR
NOP
CLRB @CSR
MOV #4368, @OPREG
MOV #2307, @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV #000, R0
MTFS R0
RTS PC

#BYTE COUNT
#PRI=7
#BUS ADDRESS
2'S COMP COUNT
#COUNT
#080 COMMAND
#DMA GO
#INTERFACE TIME
#DMA ON
#ERROR IF HERE
#081 COMMAND
#ADDRESS OF WRITE TOKEN
#BYTE COUNT
#FORMAT FOR BLUI
#DMA BYTE COUNT
#FIRE DMA
#DELAY
#GOOD DMA
#ERROR IF NOT
#CLEAR DONE
#MODSTAT
#BUFFERS FULL
#GOOD WRITE
#NO LOOP UNTIL
#CLEAR DONE BIT
#PRI=0
#LOWER CPU

```

*****
***** PROCEDURE READ BUFFER POINTER *****
*****
1 5 002274 017500 000002
2 6 002300 022700 000000
3 7 002304 001004
4 8 002306 012700 010410
5 9 002312 000167 000046
6 10 002316 022700 000001
7 11 002322 001004
8 12 002324 012700 010610
9 13 002330 000167 000030
10 14 002334 022700 000002
11 15 002340 001004
12 16 002342 012700 010510
13 17 002346 000167 000012
14 18 002352 022700 000004
15 19 002356 001020
16 20 002360 012700 010710
17 21 002364 010037 172416
18 22 002370 012737 001400
19 23 002376 105737 172416
20 24 002402 100375
21 25 002404 013700 172416
22 26 002410 042700 177400
23 27 002414 010075 000004
24 28 002420 000207
*****
RDPNT: MOV @ARG1(R5), R0
CMP #0, R0
BNE 1$
MOV #4360., R0
JMP 4$
CMP #1, R0
BNE 2$
MOV #4488., R0
JMP 4$
CMP #2, R0
BNE 3$
MOV #4424., R0
JMP 4$
CMP #4, R0
BNE 5$
MOV #4552., R0
CMP #0, @R0PREG
MOV #768., @R0PREG
TSTB @RCSR
RFL -4
MOV @#IOBUF, R0
BIC #177400, R0
MOV R0, @ARG2(R5)
RTS
*****
1$:
2$:
3$:
4$:
5$:
*****
FETCH COMMAND
INBUFO ?
NO
RDBUFADR IN0
INBUF1 ?
NO
RDBUFADR IN1
OUTBUFO ?
NO
RDBUFADR OUT0
OUTBUF1 ?
NO RETURN
RDBUFADR OUT1
RDBUFADR
RD
READY
NO LOOP UNTIL
FETCH POINTER
CLEAR MST
RETURN POINTER
*****

```

FDM.MACRO
NODE21

MACRO V03.01 4-JUN-79 17:09:17 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****

```

```

;CALL RSTART(0) -CAUSES A SOFTWARE HALT
;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
;CALL RSTART(2) -LOAD MODE(173000)

```

9	002422	017500	000002	RSTART: MOV	@R01(R5), R0				
10	002426	022700	000000	RS0: CMP	#0, R0				
11	002432	001002		BNE	RS1				
12	002434	000000		HALT					
13	002436	000207		RTS	PC				
14	002440	022700	000001	CMP	#1, R0				
15	002444	001003		BNE	RS2				
16	002446	013700	000040	MOV	@#40, R0				
17	002452	000110		JMP	(R0)				
18	002454	022700	000002	CMP	#2, R0				
19	002460	001003		BNE	RST				
20	002462	013700	002472	MOV	@#R0M, R0				
21	002466	000110		JMP	(R0)				
22	002470	000207		RTS	PC				
23	002472	173000		RDM:	.WORD	173000			
24									
25									
26									


```

1 ***** .ENABLE LSB *****
2 *****
3 ***** PROCEDURE GATE WAY INTERFACE *****
4 *****
5 *****
6
7 002474 010046      MOV     R0,-(SP)
8 002476 010146      MOV     R1,-(SP)
9 002500 010246      MOV     R2,-(SP)
10 002502 042737      BIC     #100,@#HRCSCR
11 002510 005000      CLR     R0
12 002512 012701      MOV     #XINBF,R1
13          CLR     PXOR
14 002516 105737      TSTB   @#HRCSCR
15 002522 100375      BPL     -4
16 002524 113702      MOVB   @#HRBUF,R2
17 002530 042702      BIC     #177400,R2
18          XOR     R2,PXOR
19 002534 110221      MOVB   R2,(R1)+
20 002536 005200      INC     R0
21 002540 122702      CMPB   #003,R2
22 002544 001364      BNE     2$
23 002546 022700      CMP     #6,R0
24 002552 003361      BGT     2$
25          TSTB   @#HRCSCR
26          BPL     -4
27          MOVB   @#HRBUF,R2
28          BIC     #177400,R2
29          CMFB   R2,PXOR
30          BNE     4$
31          TSTB   @#HXCSR
32          BPL     -4
33          MOVB   #006,@#HXBUF
34 002554 010067      MOV     R0,IOfLG
35          BR      5$
36          TSTB   @#HXCSR
37          BPL     -4
38          MOVB   #006,@#HXBUF
39 002560 052737      BIS     #100,@#HRCSCR
40 002566 012602      MOV     (SP)+,R2
41 002570 012601      MOV     (SP)+,R1
42 002572 012600      MOV     (SP)+,R0
43 002574 000002      RTI
44
45
46
47
48 002576 012700      MOV     #340,R0
49 002602 106400      MTPS   R0
50 002604 017500      MOV     @ARG1(R5),R0
51 002610 012701      MOV     #XOUTBF,R1
52 002614 042737      BIC     #100,@#HRCSCR
53 002622 105737      TSTB   @#HXCSR
54 002626 100375      BPL     -4
55 002630 012737      MOV     #005,@#HXBUF
56 002636 105737      TSTB   @#HRCSCR
57 002642 100375      BPL     -4

```


FDM.MACRO MACRO V03.01 4-JUN-79 17:09:17 PAGE 11-1

NODE21

58	002644	113702	177562	MOVE	@#HRCBUF, R2	
59	002650	005002		CLR	R2	
60	002652	112102		MOVE	(R1)+, R2	!FETCH BYTE
61	002654	042702	177400	BIC	#177400, R2	!READY TO SEND
62	002660	105737	177564	TSTR	@#HXCSR	!LOOP UNTIL
63	002664	100375		BFL	.-4	!SEND THE BYTE
64	002666	110237	177566	MOVE	R2, @#HXBUF	
65	002672	000240		NOF		
66	002674	000240		NOF		!SEND ALL BYTES
67	002676	077013		SOR	R0, 6\$!LOWER PRI
68	002700	012700	000000	MOV	#000, R0	
69	002704	104400		MTFS	R0	
70	002706	052737	000100 177560	BIS	#100, @#HRCSR	
71	002714	000207		RTS	PC	!ALL DONE
72	002716	000000		FXOR:	.WORD 0	
73	002720			DUMP:	.BLKW 50.	

124

```
FDM,MACRO      MACRO V03.01 4-JUN-79 17:09:17 PAGE 12
NODE21

1 *****
2 *****
3 ***** COMMON DATA AREA *****
4 *****
5 .PSECT  IFM,RW,D,GBL,REL,OVR
6
7 OUTBF:  .BLKB  256.
8 INBF:   .BLKB  256.
9 IURTTM: .BLKW
10 IURTI:  .BLKW
11 STAT:   .BLKW
12 LTIME:  .BLKW
13 .EVEN
14
15 .PSECT  BUFS,RW,D,GBL,REL,OVR
16
17 XOUTBF: .BLKB  256.
18 XINBF:  .BLKB  256.
19 ACKSEQ: .BLKB  256.
20 LIDFD:  .BLKB
21 IOFLG:  .BLKW
22 LLFLG:  .BLKW
23 IRSEND: .BLKW
24 .EVEN
25
26
27 000001 .END
```

FDM-MACRO SYMBOL TABLE MACRO V03.01 4-JUN-79 17:09:17 PAGE 12-1

ACKSEQ	001000R	003	B1\$	000730R	E0\$	001234R	LFINPT=	***** G	RT1\$	001006R	002
AREA	001214R		B2\$	000516R	E1\$	001244R	LPOUT	002110RG	STAT	001004R	
ARG1	= 000002		B3\$	000542R	G10	002474RG	LTIME	001006R	002	STAT0	001430RG
ARG2	= 000004		B4\$	000566R	GOUT	002576RG	MASTER	000000RG	STATUS	001332RG	
ARG3	= 000006		B5\$	000614R	G4\$	002446R	NDX\$	= 000000	STAT\$	000412R	
ARG4	= 000010		B6\$	000632R	HRBUF	= 177562	ND24	= 000001	SWITCH	001274RG	
BAR	= 172410		B7\$	000576R	HRCR\$	= 177560	NHOST	= 000000	TIME	001662RG	
B100	= 000001		CAUSE	001212R	HXBUR	= 177566	OPREG	= 172416	WCR	= 172412	
B101	= 000002		CRCK	001062R	HXC\$R	= 177564	OUTBF		002	WTD	001660R
B102	= 000004		CSR	= 172414	INBF	000400R	002	PXOR	WTKG	001560R	
B103	= 000010		DATA	001220R	IOBUR	= 172416	KAM		WTKEN	001502RG	
B104	= 000020		DESTR	001264RG	IOFLG	002000R	003	RDPNT	WTRAM	001716R	
B105	= 000040		DHOST	= 000001	IRSEND	002004R	003	RDRAM	XCIO	= 000001	
B106	= 000100		DMADK	001166R	IURT	001002R	002	ROM	XCIO	= 000000	
B107	= 000200		DUMP	002720R	IMRTM	001000R	002	RST	XINBF	000400R	003
B108	= 000400		EMBF	001032R	LIDFD	001400R	003	RSTART	XOUTBF	000000R	003
B114	= 040000		ENABLE	001232RG	L10	000436RG	RS0		XSIO	= 000001	
B115	= 100000		END\$	001210R	LIJINT	000104RG	RS1		ZEROBP	000312R	
B0\$	000652R		ENSTR	001254RG	LLFLG	002002R	003	RS2			

. ABS. 000000 000
 DFM 003064 001
 DFM 001010 002
 BUFS 002006 003
 ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 51 PAGES
 ,DK:FDM=DK1FDM

PAGE 001

12:00:00

01-JUL-79

CONF21.COM

PROGRAM:

decurter

FOR NORMAL/NO LINE
FOR NORMAL 1/NO LINE
MAC FID

PAGE 001

12:00:00

01-JUL-79

LNK21.COM

PROGRAM:

decompiler

ASSIGN DX0: DN:

R LINK

DX1:NODE21.SAV,DX1:MAP=DX1:NODAL/C/I/W

DX1:FDM/C

DX1:NODAL1//

\$SIMRT

ASSIGN DX1: DX:

1.3 NODE 22 (VSQC)

The Voice Service Quality Control (VSQC) module assesses the performance of voice channels for the purpose of detecting degrading performance and assisting in fault isolation. The MSCDM is capable of monitoring at least 1000 channels. By convention, channels numbered 1-500 are monitored by the VSQC. There are six parameters to be checked per channel.

1. Peak Power (PK)
2. Average Power (AV)
3. Frequency Offset (FO)
4. Phase Jitter (PJ)
5. C-Message Noise (CN)
6. 3 KHz Flat Noise (FN)

Values for these parameters are generated by the Simulated Input Generator (SIG) microcomputer. Thresholding is done on the parameters by comparison with the Red High (RH), Red Low (RL), Amber High (AH), and Amber Low (AL) values in order to determine whether the channel parameters are in the Red, Amber or Green region.

VSQC Parameter Thresholds

<u>Parameter</u>	<u>RH</u>	<u>RL</u>	<u>AH</u>	<u>AL</u>
PK	10	0	9	2
AV	10	0	8	2
FO	20	-100	0	-80
PJ	10	-90	-10	-70
CN	0	-80	-10	-60
FN	20	-100	0	-80

The VSQC also performs trending on the parameter values. Trending begins when the parameter is within a delta value of the Threshold. The critical trending values for the VSQC parameter thresholds are given below.

VSQC Critical Trending Values

<u>Parameter</u>	<u>Delta</u>	<u>RHT</u>	<u>RLT</u>	<u>AHT</u>	<u>ALT</u>
PK	0.5	9.5	0.5	7.5	2.5
AV	0.5	9.5	0.5	7.5	2.5
FO	3	17	-97	-3	-77
PJ	2	8	-88	-8	-68
CN	1	-1	-79	-19	-59
FN	2	18	-98	-2	-78

The threshold and trending values may be modified by modifying the program DATA statement and recompiling the VSQC nodal software.

Event Reports are sent to the FIAC when a parameter value is Amber or Red and the Event Reporting condition parameter is turned ON (via Module Update Mode 3 of the User Language). The Event Report consists of an 8-byte trunk name, 4-byte channel number, one-byte condition, one-byte parameter number that caused the Red or Amber condition, 3-byte monitor point number, and the node designator to which reports should be sent.

The VSQC interprets command from the DBMS generated by Mode 3 of the User Language. The commands include Event Reporting ON or OFF; if ON, the node designator to which the Event Reports are to be sent, i.e., the terminal used for display, is given. The VSQC also interprets the command, received as a packet from the loop, to measure a specific channel which is sent to the SIG; the resulting measurement value is sent to the OCRI terminal requesting the measurement (via Mode 3 of the User Language).

1.3.1 Program Descriptions

1.3.1.1 Refer to Section 1.1 for descriptions of routines NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME RAM, LPOUT, RDPNT, RSTART.

1.3.1.2 Subroutine VSQC (FORTRAN)

This subroutine is called from the nodal program when a packet has been received from the SIG interface. It decodes the simulated input values read from the SIG and performs the function described in Section 1.3.

1.3.1.3 Subroutine TREND (FORTRAN)

This subroutine is called by the VSQC routine to perform the trending on the values sent by the SIG. This value is compared with DELTA values; when a channel consistently is in the DELTA region, an event report is generated.

1.3.1.4 Subroutine VSQLP (FORTRAN)

The subroutine is called from program NODAL when a packet is received from the loop. It performs one of three functions: turns reporting on or off, and takes a measurement of a specified channel.

1.3.1.5 Subroutine SIO (MACRO)

The subroutine is called from an interrupt received by the SIG interface; it then reads the message and sets a flag containing the byte count of the message.

1.3.1.6 Subroutine SOUT (MACRO)

This subroutine is called from subroutine VSQC and passes to the SIG processor the channel number of a channel for which the user language requests a measurement.

1.3.1.7 Subroutine READY (MACRO)

This subroutine is called to signal the SIG that VSQC is ready for another measurement.


```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:27:42      PAGE 001

0001      PROGRAM NODAL
0002      INTEGER*2 XING,XOUTQ,ACKQ,FXING,FXOUTQ,PACKQ
0003      INTEGER*2 PINQ,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INRF
0007      LOGICAL*1 IDATA
0008      LOGICAL*1 PACK,ICODE,ETX,CR,LF,MONITOR,ISLID,DUM
0009      INTEGER*2 LTIME,ACKTIM,NEWTIM,OLDTIM,IWRITM,TIMLIM,ATIMLM
0010      REAL*4 RH,RL,AH,AL,RHT,RLT,AHT,ALT,DELTA,IRSV,UHEAS
0011      REAL*8 VTR
0012      COMMON /MESS/ MESSEQ
0013      COMMON /DFM/ OUTBF(256),INRF(256),
0014      1 IWRITM,IWRT,STAT,LTIME
0015      1 COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),FXOUTQ(2),
0016      1 ACKQ(16),PACKQ(2),ING(16),PING(2)
0017      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0018      1 LIDFD(256),IOFLG,LLFLG,IRSEND
0019      1 COMMON /FRE/ FREE(64),IFR,IFRSZ
0020      1 COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0021      1 COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0022      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0023      1 RESNLM,IALT,ISLID,MONITOR,LF,ETX,CR,DUM
0024      1 COMMON /VCOM/ ISWCH,VMEAS(6),IRSV(6),ICTR,RL(6),RH(6),ICHNSW,
0025      1 AL(6),AH(6),DELTA(6),ITER,NFL,SFL,VTR,IVC,MON,IDATA(10)
0026      CALL MASTER
0027      CALL INIT
0028      CALL ENABLE(1)
0029      5 CALL READY
0030      25 IF(PING(1) .LT. PING(2)) GOTO 40
0031      CALL ENABLE(0)
0032      Q2=DEQUE(PING,ING,1)
0033      CALL ENABLE(1)
0034      DO 30 I=1,10
0035      IDATA(I)=PACK(I+6,Q2)
0036      30 CONTINUE
0037      CALL VSOCLP
0038      IFR=IFR+1
0039      FREE(IFR)=Q2
0040      CALL STATB0(15)
0041      IF(15 .EQ. 1) GOTO 100
0042      IF(10FLG .GT. 0 .AND. IRSEND .EQ. 0) CALL VSOCLP
0043      IF(11FLG .GT. 0) CALL LINLOS
0044      IF(OUTFCT .EQ. 1) GOTO 100
0045      IF((IFULL .EQ. 1) .OR. (ISENT .EQ. 1)) GOTO 100
0046      IF(IRSEND .EQ. 1) GOTO 85
0047      IF(PXING(1) .LT. PXING(2)) GOTO 100
0048      CALL ENABLE(0)
0049      Q1=DEQUE(PXING,XING,1)
0050      CALL ENABLE(1)
0051      OUTQ=Q1
0052      85 CALL DESTR(PACK(254,OUTQ),Q2)
0053      DO 90 I=1,Q2
0054

```

PAGE 002

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:27:42

```

0058   OUTBF(I)=PACK(I,OUTQ)
0059   CONTINUE
0060   OUTBF(Q2+1)=0
0061   IPT=OUTBF(5)
0062   OUTBF(Q2+2)=LIBFD(IPT)
0063   IFULL=1
0064   ISENT=1
0065   CALL LPOUT(Q2+2)
0066   ISEND=1
0067   ACKTM=0
0068   IWRTM=0
0069   IWRT=1
0070   INFLCT=0
0071   OLDTM=NEWTIM
0072   NEWTM=LTIME
0073   IF(IWRT.EQ.0) GOTO 120
0075   IWRTM=IWRTM+(NEWTIM-OLDTM)
0076   IF(IWRTM.LT. TIMLM) GOTO 120
0078   CALL WTOKEN
0079   IWRTM=0
0080   IWRT=0
0081   IF(ISENT.EQ.0) GOTO 130
0083   ACKTM=ACKTM+(NEWTIM-OLDTM)
0084   IF(ACKTM.LT. ATIMLM) GOTO 130
0086   CALL ENABLE(0)
0087   CALL ACKNAK(0)
0088   CALL ENABLE(1)
0089   CONTINUE
0090   GOTO 5
0091   END

```

PAGE 001

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:28:03

```

0001 FUNCTION IGETSP(N)
0002 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003 INTEGER*2 FREE,FLWCNT
0004 INTEGER*2 OUTFCT,OUTQ,RESNLM
0005 COMMON /FRE/ FREE(64),IFR,IFRSZ
0006 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1 RESNLM,IALTRY,ISLID,MONTOR,LF,ETX,CR,DUM
0007 IF (IFR .LT. 1) CALL INIT
0008 IGETSP=FREE(IFR)
0009 IFR=IFR-1
0010 RETURN
0011 END
0012

```

```
FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:28:15      PAGE 001

0001      SUBROUTINE ENQUE(A,B,N)
0002      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003      INTEGER*2 XING,PXING,XOUTQ,PXOUTQ,ACKQ,PACKQ,ING,PING
0004      INTEGER*2 FLWCNT,OUTQ,A(2),B(16)
0005      INTEGER*2 RESNLM,OUTFCT
0006      COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
0007      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1 RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0009      10 IQHEAD=A(1)
0010      IQTAIL=A(2)
0011      IF(IQTAIL.EQ.1) GOTO 20
0012      IQTAIL=IQTAIL-1
0013      B(IQTAIL)=N
0014      A(2)=IQTAIL
0015      GOTO 999
0016      20 IF(IQHEAD.GE.(IQLNTH)) GOTO 40
0017      NN=IQHEAD-IQTAIL
0018      DO 30 I=1,NN+1
0019      30 B(IQLNTH+1-I)=B(IQHEAD+1-I)
0020      A(1)=IQLNTH
0021      A(2)=IQLNTH-NN
0022      GOTO 10
0023      40 CALL INIT
0024      999 RETURN
0025      END
0026
```

```
FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:28:28      PAGE 001

0001      FUNCTION DEQUE(A,B,IO)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOR
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACNQ,FACNQ,INQ,PINQ
0004      INTEGER*2 FLWCNT,A(2),B(16),DEQUE,OUTFCT,OUTQ,RESNLM
0005      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      ACNQ(16),FACNQ(2),INQ(16),PINQ(2)
0007      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,IOLNTH+1,IFULL,OUTQ,
0008      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0009      IOHEAD=A(1)
0010      DEQUE=B(IOHEAD)
0011      IF(ID.NE.1) GOTO 999
0012      IF(IOHEAD.NE.0) GOTO 10
0013      A(1)=IOLNTH
0014      A(2)=IOLNTH+1
0015      GOTO 999
0016      10 A(1)=IOHEAD-1
0017      999 RETURN
0018      END
```



```
FORTTRAN IV      V02.1-1      Mon 04-Jun-79 17:28:50      PAGE 001

0001      SUBROUTINE ACKNAK(N)
0002      INTEGER*2 FLWCNT,OUTFCT,STAT,XINQ,PXINQ,PXOUTQ
0003      INTEGER*2 ACKQ,PACKQ,PINQ,FREE,OUTQ,XOUTQ
0004      INTEGER*2 I1,I2,I3,I4,I5,RESNLH
0005      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0006      LOGICAL*1 PACK,OUTBF,INBF,LCOMT(40)
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0008      INTEGER*2 LTIME,IWRTH
0009      REAL*8 COMT(5)
0010      EQUIVALENCE(COMT,LCOMT)
0011      DATA COMT(1)/MESSAGE //,COMT(2)/NOT SENT//,COMT(3)/FROM //,
0012      1 COMT(4)/NODE //,COMT(5)/TO NODE//
0013      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0014      1 RESNLH,IALRT,ISLID,MONTOR,LF,ETX,CR,DUM
0015      COMMON /DFM/ OUTBF(256),INBF(256),
0016      1 IWRTH,IWRT,STAT,LTIME
0017      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0018      1 PACKQ(2),INQ(16),PINQ(2)
0019      COMMON /FRE/ FREE(64),IFR,IFRSZ
0020      COMMON /MESS/ MESSEQ
0021      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0022      1 IDFLG,LLFLG,IRSEND
0023      IF (N.NE. 1) GOTO 10
0024      IFR=IFR+1
0025      DO 5 I=1,256
0026      5 CONTINUE
0027      IFULL=0
0028      ISENT=0
0029      IRSEND=0
0030      FREE(IFR)=OUTQ
0031      GOTO 999
0032      T2=PACK(256,OUTQ)
0033      IF (T2 .LE. RESNLH) GOTO 120
0034      T3=PACK(3,OUTQ)
0035      T4=MOD(T3,64)
0036      IF (T4 .LT. 32) GOTO 110
0037      T5=IGETSP(N)
0038      IF (MESSEQ .EQ. 126) MESSEQ=0
0039      MESSEQ=MESSEQ+1
0040      PACK(1,I5)=0
0041      PACK(2,I5)=MESSEQ
0042      PACK(3,I5)=0
0043      PACK(4,I5)=0
0044      PACK(5,I5)=25
0045      PACK(6,I5)=ISLID
0046      DO 20 I=7,60
0047      20 CONTINUE
0048      PACK(I,I5)=*040
0049      DO 30 I=7,9
0050      30 CONTINUE
0051      PACK(I,I5)=LF
0052      30 CONTINUE
0053
```

EDM.M
SYMB
ACKSE
AREA
ARG
ARG2
ARG3
ARG4
BAR
BIT00
BIT01
BIT02
BIT03
BIT04
BIT05
BIT06
BIT07
BIT08
BIT14
BIT15
B0\$
ABS
DFM
BUFS
ERROR
VIRTU
DYNAM
'DKIF

62

PAGE 002

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:28:50

```

0054 DO 40 I=1,8
0055   PACK(I+9,T5)=LCOMT(I)
0056   CONTINUE
0057 DO 50 I=9,16
0058   PACK(I+9,T5)=LCOMT(I)
0059   CONTINUE
0060 DO 60 I=17,22
0061   PACK(I+9,T5)=LCOMT(I)
0062   CONTINUE
0063 DO 70 I=23,29
0064   PACK(I+7,T5)=LCOMT(I)
0065   CONTINUE
0066 ENCODE(3,80,PACK(38,T5))ISLID
0067 FORNAT(13)
0068 DO 90 I=33,40
0069   PACK(I+11,T5)=LCOMT(I)
0070   CONTINUE
0071 ENCODE(3,80,PACK(52,T5))PACK(5,OUTQ)
0072   PACK(55,T5)=LF
0073   PACK(56,T5)=LF
0074   PACK(57,T5)=CR
0075   PACK(58,T5)=ETX
0076   CALL ENSTR(PACK(254,T5),58)
0077   CALL ENQUE(PXING,XING,T5)
0078 DO 102 I=1,256
0079   PACK(I,OUTQ)=0
0080   CONTINUE
0081   ISENT=0
0082   IRSEND=0
0083   IFULL=0
0084   IWR=0
0085   IFR=IFR+1
0086   FREE(IFR)=OUTQ
0087   GOTD 999
0088   110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089   PACK(256,OUTQ)=0
0090   ISENT=0
0091   IFULL=0
0092   IF(IRSEND.EQ.1) GOTD 999
0093   CALL ENQUE(PXING,XING,OUTQ)
0094   GOTD 999
0095   120 PACK(256,OUTQ)=T2
0096   IFULL=0
0097   ISENT=0
0098   IF(IRSEND.EQ.1) GOTD 999
0099   CALL ENQUE(PXING,XING,OUTQ)
0100   RETURN
0101   999
0102   END
0103

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:29:11      PAGE 001

0001      SUBROUTINE INPTQ(L)
0002      INTEGER*2 STAT,XING,PXING,XOUTQ,PXOUTQ,ACKQ
0003      INTEGER*2 PACKQ,PING,FLWCNT,OUTFCT,T1,RESNLM,OUTQ
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0005      LOGICAL*1 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
0006      INTEGER*2 LTIME,IWRITM
0007      COMMON /DFM/ OUTBF(256),INBF(256),
0008      1 COMMON IWRITM,IWRT,STAT,LTIME
0009      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010      LIDFD(256),IOFLG,LLFLG,IRSEND
0011      1 COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
0012      ACKQ(16),PACKQ(2),ING(16),PING(2)
0013      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0014      RESNLM,IALTRI,ISLID,MONITOR,LF,ETX,CR,DUM
0015      IF(L.LT. 1) GOTO 30
0016      LI=L-2
0017      II=INBF(6)
0018      IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
0019      T1=IGETSP(N)
0020      DO 10 I=1,LI
0021      10 PACK(I,T1)=INBF(I)
0022      CALL ENSTR(PACK(254,T1),LI)
0023      CALL ENQUE(PING,ING,T1)
0024      N=2
0025      GOTO 40
0026      N=1
0027      GOTO 40
0028      N=0
0029      DO 50 I=1,2
0030      50 OUTBF(I)=INBF(I)
0031      CONTINUE
0032      NN=1
0033      IF(N.EQ. 0) NN=128
0034      OUTBF(3)=NN
0035      OUTBF(4)=0
0036      OUTBF(5)=INBF(6)
0037      OUTBF(6)=ISLID
0038      OUTBF(7)=ETX
0039      OUTBF(8)=0
0040      OUTBF(9)=LIDFD(INBF(6))
0041      CALL LPOUT(9)
0042      IF(N.NE. 2) GOTO 999
0043      ACKSEQ(II)=INBF(2)
0044      999 RETURN
0045      END
0046

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:29:28      PAGE 001

0001      SUBROUTINE LPIIPT(LI)
0002      INTEGER*2 STAT,FLWCNT,OUTFCT,OUTG,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONITOR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRTM
0006      COMMON/DPH/OUTBF(256),INBF(256),IWRTM,IWRT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,FLWCNT,IGLNTH,OUTFCT,IFULL,OUTG,
1          RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0008      CC1=INBF(3)
0009      CC2=INBF(4)
0010      IF(CC1.LT. 0) GOTO 25
0012      IF(CC1.EQ. 0 .AND. CC2.EQ. 0) GOTO 40
0014      CC3=MOD(CC1,2)
0015      CC4=MOD(CC1,256)
0016      IF ((CC3.GE. 1 .OR. CC4.GE. 128) .AND. ISENT.EQ. 1) GOTO 20
0018      IF ((INBF(1).EQ. 85) .AND. INBF(2).EQ. 170) GOTO 99
0020      CC5=MOD(INBF(3),64)
0021      IF(CC5.GE. 32) CALL INPTQ(LI)
0023      GOTO 99
0024      CC7=MOD(CC1,2)
20      IF(CC7.GE. 1) GOTO 30
0025      CALL ACKNAK(0)
25      GOTO 99
0028      GOTO 99
30      CALL ACKNAK(1)
0029      GOTO 99
0030      GOTO 99
40      CALL INPTQ(LI)
99      CONTINUE
0032      RETURN
0033      END
0034

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:29:42      PAGE 001

0001      SUBROUTINE INIT
0002      REAL*4 RH,RL,AH,AL,IRSV,VMEAS
0003      REAL*8 VTR
0004      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ,FREE
0005      INTEGER*2 STAT,FLWCNT,OUTFCT
0006      INTEGER*2 SETPRM,RSTPRM,SETBKP,OUTQ,RESNLM,RSTBKP
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0008      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID,DUM,IDATA
0009      INTEGER*2 LTIME,OLDTIM,TIMLIM,ACKTIM,ATIMLM,IWRTIM
0010      COMMON /DFM/ OUTBF(256),INBF(256),
0011      1 IWRTIM,IWRT,STAT,LTIME
0012      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0013      1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0014      COMMON /MESS/ MESSEQ
0015      COMMON /RUF/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0016      1 IOFLG,LLFLG,IRSEND
0017      COMMON /FRE/ FREE(64),IFR,IFRSZ
0018      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0019      COMMON /SMT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0020      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0021      1 RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0022      COMMON /UCOM/ ISWITCH,VMEAS(6),IRSV(6),ICTR,RL(6),RH(6),ICHNSW,
0023      1 AL(6),AH(6),DELTA(6),ITERM,NFL,SFL,VTR,IVC,MON,IDATA(10)
0024      CALL RAM(0,3,4)
0025      CALL RAM(0,255,0)
0026      MESSEQ=0
0027      IOFLG=0
0028      IRSEND=0
0029      LLFLG=0
0030      IFRSZ=64
0031      INFLCT=0
0032      ISLID=22
0033      CR=.015
0034      LF=.012
0035      ETX=.003
0036      ICHNSW=0
0037      RESNLM=2
0038      IQLNTH=16
0039      IALTRT=0
0040      FLWCNT=0
0041      IWRTIM=0
0042      TIMLIM=50
0043      SETPRM=4
0044      RSTPRM=32
0045      SETBKP=8
0046      RSTBKP=64
0047      ATIMLM=200
0048      MONTOR=27
0049      IWRT=0
0050      ISWITCH=0
0051      OUTFCT=0
0052      IFULL=0
0053      ISENT=0

```


PAGE 002

Mon 04-Jun-79 17:29:42

V02.1-1

FORTRAN IV

```

0050      DO 40 I=1,20
0051      LIDFD(I)=4
0052      LIDFD(21)=1
0053      LIDFD(22)=3
0054      LIDFD(23)=6
0055      LIDFD(24)=5
0056      LIDFD(25)=7
0057      LIDFD(26)=8
0058      LIDFD(27)=9
0059      LIDFD(28)=2
0060      DO 50 I=29,39
0061      LIDFD(I)=0
0062      DO 60 I=40,44
0063      LIDFD(I)=1
0064      DO 70 I=45,59
0065      LIDFD(I)=0
0066      DO 80 I=60,64
0067      LIDFD(I)=2
0068      DO 90 I=65,79
0069      LIDFD(I)=0
0070      DO 100 I=80,84
0071      LIDFD(I)=4
0072      DO 110 I=85,99
0073      LIDFD(I)=0
0074      DO 120 I=100,104
0075      LIDFD(I)=5
0076      DO 130 I=105,256
0077      LIDFD(I)=0
0078      PING(1)=IQLNTH
0079      PACKQ(1)=IQLNTH+1
0080      PACKQ(2)=IQLNTH+1
0081      PXOUTQ(1)=IQLNTH
0082      PXOUTQ(2)=IQLNTH+1
0083      PING(1)=IQLNTH
0084      PING(2)=IQLNTH+1
0085      DO 10 I=1,IFRSZ
0086      PACK(255,I)=0
0087      PACK(256,I)=0
0088      DO 20 I=1,IFRSZ
0089      FREE(I)=1
0090      DO 30 I=1,256
0091      ACKSEQ(I)=256
0092      IFR=IFRSZ
0093      RETURN
0094      END
0095

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:30:04      PAGE 001

0001      SUBROUTINE LINLOS
0002      REAL*8 RM1(5),LIN08,LIN18
0003      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM,T1
0004      INTEGER*2 XING,PXING,XOUTQ,ACKQ,PACKQ,PING
0005      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0006      LOGICAL*1 ETX,CR,LF,MONTR,ISLID,DUM
0007      LOGICAL*1 PACK,M1(40),LIN0(8),LIN1(8)
0008      COMMON/DUE/XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
          &      ACKQ(16),PACKQ(2),INQ(16),PING(2)
0009      COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
          &      IOFLG,LLFLG,IRSEND
0010      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
          &      RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
          &      PACK(256,64)
0011      COMMON /MESS/ MESSEQ
0012      DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0013      DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0014      DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0015      DATA RM1(5)/'DE 22 '//
0016      EQUIVALENCE(LIN0,LIN08)
0017      EQUIVALENCE(LIN1,LIN18)
0018      EQUIVALENCE(M1,RM1)
0019      IS=LLFLG
0020      CALL ENABLE(0)
0021      T1=IGETSP(N)
0022      CALL ENABLE(1)
0023      PACK(1,T1)=0
0024      IF(MESSEQ.EQ.126) MESSEQ=0
0025      MESSEQ=MESSEQ+1
0026      PACK(2,T1)=MESSEQ
0027      PACK(3,T1)=0
0028      PACK(4,T1)=0
0029      PACK(5,T1)=25
0030      PACK(6,T1)=ISLID
0031      DO 20 I=7,9
0032      PACK(I,T1)=LF
0033      PACK(I,T1)=LF
0034      20 CONTINUE
0035      DO 30 I=1,22
0036      PACK(I+9,T1)=M1(I)
0037      IF(IS.EQ.1) GOTO 50
0038      DO 40 I=1,8
0039      PACK(I+31,T1)=LIN0(I)
0040      GOTO 70
0041      DO 60 I=1,8
0042      PACK(I+31,T1)=LIN1(I)
0043      DO 80 I=23,40
0044      PACK(I+17,T1)=M1(I)
0045      PACK(58,T1)=CR
0046      PACK(59,T1)=LF
0047      PACK(60,T1)=ETX
0048      CALL ENSTR(PACK(254,T1),60)
0049      CALL ENABLE(0)
0050      CALL ENQUE(PXING,XING,T1)
0051      CALL ENABLE(1)
0052
0053

```

PAGE 002

Mon 04-Jun-79 17:30:04

FOTRAN IV U02.1-1

0054 LLFLG=0
0055 RETURN
0056 END

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 18:11:26      PAGE 001

0001      SUBROUTINE VSQC
0002      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ
0003      INTEGER*2 PINQ,FLWCNT,OUTFCT,OUTQ,RESNLM
0004      REAL*8 VTR
0005      REAL*4 RH,RL,AH,AL,RHT,RLT,AHT,ALT,DELTA,IRSV,VMEAS
0006      LOGICAL*1 PACK,ISLID,MONTR,LF,ETX,CR,DUM,INQUE
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,IData
0008      COMMON PACK(256,64)
0009      COMMON /MESS/ MESSEQ
0010      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0011      1 IOFLG,LLFLG,IRSEND
0012      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0013      1 PACKQ(2),INQ(16),PINQ(2)
0014      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,RESNLM,
0015      1 IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0016      DATA RH(1),RH(2),RH(3),RH(4) /99.5,99.5,199.0,199.0/
0017      DATA RL(1),RL(2),RL(3),RL(4) /-90.5,-90.5,-270.0,-275.0/
0018      DATA AL(1),AL(2),AL(3),AL(4) /90.0,90.0,192.0,191.5/
0019      DATA AH(1),AH(2),AH(3),AH(4) /56.5,199.0/
0020      DATA AL(1),AL(2),AL(3),AL(4) /-50.0,-55.0,-251.0,-257.0/
0021      DATA AL(5),AL(6) /-113.5,-252.0/
0022      DATA DELTA(1),DELTA(2),DELTA(3) /5.5,5.3,0/
0023      DATA DELTA(4),DELTA(5),DELTA(6) /2.0,1.0,2.0/
0024      DECODE(2,20,XINBF(1),ERR=410) ICNT
0025      DECODE(4,30,XINBF(3),ERR=410) VMEAS(1)
0026      DECODE(4,30,XINBF(7),ERR=410) VMEAS(2)
0027      DECODE(4,30,XINBF(11),ERR=410) VMEAS(3)
0028      DECODE(4,30,XINBF(15),ERR=410) VMEAS(4)
0029      DECODE(4,30,XINBF(19),ERR=410) VMEAS(5)
0030      DECODE(4,30,XINBF(23),ERR=410) VMEAS(6)
0031      DECODE(2,20,XINBF(27),ERR=410) IVC
0032      DECODE(2,20,XINBF(37),ERR=410) MON
0033      FORMAT(A2)
0034      30 FORMAT(A4)
0035      IF(ICHNSW .EQ. 1) GOTO 400
0036      DO 10 I=1,6
0037      10 IRSV(I)=(AH(I)+AL(I)) / 2
0038      CONTINUE
0039      DO 350 ICTR=1,6
0040      ALT=AL(ICTR)+DELTA(ICTR)
0041      AHT=AH(ICTR)+DELTA(ICTR)
0042      RLT=RL(ICTR)+DELTA(ICTR)
0043      RHT=RH(ICTR)+DELTA(ICTR)
0044      IF(VMEAS(ICTR) .GE. ALT .AND.
0045      1 VMEAS(ICTR) .LE. AHT) GOTO 350
0046      IF((VMEAS(ICTR) .GT. AL(ICTR) .AND.
0047      1 VMEAS(ICTR) .LT. ALT) .OR.
0048      2 (VMEAS(ICTR) .GT. AHT .AND.
0049      3 VMEAS(ICTR) .LT. AH(ICTR))) GOTO 100

```

```

0049 IF((VMEAS(ICTR).GE. AH(ICTR)).AND.
1 VMEAS(ICTR).LE. RHT).OR.
2 (VMEAS(ICTR).GE. RLT).AND.
3 VMEAS(ICTR).LE. AL(ICTR))) GOTO 200
0051 IF((VMEAS(ICTR).GT. RHT).AND.
1 VMEAS(ICTR).LT. RH(ICTR)).OR.
2 (VMEAS(ICTR).GT. RLT).AND.
3 VMEAS(ICTR).LT. RL(ICTR))) GOTO 300
0053 IF(ISWITCH.EQ. 1) GOTO 45
0055 IOFLG=0
0056 RETURN
0057 45 CALL ENABLE(0)
0058 K1=IGETSP(N)
0059 CALL ENABLE(1)
0060 DO 46 I=1,256
0061 PACK(I,K1)=0
0062 46 CONTINUE
0063 IF(MESSEQ.EQ. 126) MESSEQ=0
0065 MESSEQ=MESSEQ+1
0066 PACK(1,K1)=0
0067 PACK(2,K1)=MESSEQ
0068 PACK(3,K1)=0
0069 PACK(4,K1)=0
0070 PACK(5,K1)=MONTOR
0071 PACK(6,K1)=ISLID
0072 DO 50 I=7,14
0073 PACK(I,K1)=XINBF(I+22)
0074 50 CONTINUE
0075 DO 60 I=15,16
0076 PACK(I,K1)=XINBF(I+12)
0077 60 CONTINUE
0078 PACK(17,K1)=1
0079 ENCODE(4,65,PACK(18,K1)) VMEAS(ICTR)
0080 FORMAT(A4)
0081 ENCODE(2,70,PACK(22,K1)) MON
0082 70 FORMAT(A2)
0083 PACK(24,K1)=ITERM
0084 CALL ENSTR(PACK(25,K1),24)
0085 CALL ENABLE(0)
0086 CALL ENQUE(PXINQ,XINQ,K1)
0087 CALL ENABLE(1)
0088 GOTO 410
0089 100 IF(ISWITCH.EQ. 1) GOTO 110
0091 GOTO 410
0092 110 CALL TREND(2,VMEAS(ICTR))
0093 GOTO 410
0094 200 IF(ISWITCH.EQ. 1) GOTO 210
0096 GOTO 410
0097 210 CALL TREND(2,VMEAS(ICTR))
0098 GOTO 410
0099 300 IF(ISWITCH.EQ. 1) GOTO 310
0101 GOTO 410
0102 310 CALL TREND(2,VMEAS(ICTR))
0103 GOTO 410

```



```

0104 350 CONTINUE
0105 GOTO 410
0106 400 CALL ENABLE(0)
0107 K1=IGETSP(N)
0108 CALL ENABLE(1)
0109 DO 405 I=1,256
0110 PACK(I,K1)=0
0111 CONTINUE
0112 IF(MESSED.EQ.126) MESSED=0
0113 MESSED=MESSED+1
0114 PACK(1,K1)=0
0115 PACK(2,K1)=MESSED
0116 PACK(3,K1)=0
0117 PACK(4,K1)=0
0118 PACK(5,K1)=25
0119 PACK(6,K1)=ISLID
0120 PACK(7,K1)=CR
0121 PACK(8,K1)=LF
0122 ENCODE(4,360,PACK(9,K1)) ICHAN
0123 PACK(13,K1)='040
0124 ENCODE(12,370,PACK(14,K1)) VMEAS(1)
0125 PACK(26,K1)='040
0126 ENCODE(12,370,PACK(27,K1)) VMEAS(2)
0127 PACK(39,K1)='040
0128 ENCODE(12,370,PACK(40,K1)) VMEAS(3)
0129 PACK(52,K1)='040
0130 ENCODE(12,370,PACK(53,K1)) VMEAS(4)
0131 PACK(65,K1)='040
0132 ENCODE(12,370,PACK(66,K1)) VMEAS(5)
0133 PACK(78,K1)='040
0134 ENCODE(12,370,PACK(79,K1)) VMEAS(6)
0135 PACK(91,K1)=CR
0136 PACK(92,K1)=LF
0137 PACK(93,K1)=ETX
0138 CALL ENSTR(PACK(254,K1),93)
0139 CALL ENABLE(0)
0140 CALL ENQUE(PXING,XIND,K1)
0141 CALL ENABLE(1)
0142 ICHNSW=0
0143 FORMAT(I4)
0144 360 FORMAT(F12.6)
0145 370 IOFLG=0
0146 410 RETURN
0147 END
0148

```

```

0001 SUBROUTINE TREND(ICND,IVAL)
0002 REAL*4 RH,RL,AH,AL,RHT,RLT,AHT,ALT,DELTA,IRSV,VMEAS,IVAL
0003 REAL*8 VTR
0004 LOGICAL*1 PACK,ISLID,MONITOR,LF,ETX,CR,DUM,INQUE,IData
0005 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0006 INTEGER*2 XING,PXING,XOUTQ,PXOUTQ,ACKQ,PACKQ,PING
0007 INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM
0008 COMMON PACK(256,64)
0009 COMMON /MESS/ MESSEQ
0010 COMMON /VCON/ ISWCH,VMEAS(6),IRSV(6),ICTR,RL(6),RH(6),ICHNSW,
1 AL(6),AH(6),DELTA(6),ITERM,NFL,SFL,VTR,IVC,MON,IData(10),
2 ICHAN
0011 COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
1 PACKQ(2),INQ(16),PING(2)
0012 COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,RESNLM,
1 IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0013 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1 IOFLG,LLFLG,IRSEND
0014 IU=(RH,ICTR)-RL,ICTR)/30
0015 IF((IRSV,ICTR)-IVAL).GT.(4*IU)) GOTO 20
0016 IF(NFL.GE.3) GOTO 40
0017 IF((IRSV,ICTR)-IVAL).GT.(2*IU)) GOTO 30
0018 NFL=NFL+1
0019 IRSV(ICTR)=(.33*IVAL+.67*IRSV(ICTR))
10 GOTO 60
0020 IF(SFL.GE.2) GOTO 30
0021 SFL=SFL+1
0022 GOTO 50
0023 IRSV(ICTR)=IVAL
0024 NFL=0
0025 GOTO 50
0026 SFL=0
0027 GOTO 10
0028 IF(ICND.EQ.2) RETURN
0029 CALL ENABLE(0)
0030 K1=IGETSP(N)
0031 CALL ENABLE(1)
0032 IF(MESSEQ.EQ.126) MESSEQ=0
0033 MESSEQ=MESSEQ+1
0034 PACK(1,K1)=0
0035 PACK(2,K1)=MESSEQ
0036 PACK(3,K1)=0
0037 PACK(4,K1)=0
0038 PACK(5,K1)=MONITOR
0039 PACK(6,K1)=ISLID
0040 DO 70 I=7,14
0041 PACK(I,K1)=XINBF(I+22)
0042 CONTINUE
0043 DO 75 I=15,16
0044 PACK(I,K1)=XINBF(I+12)
0045 CONTINUE
0046 ENCODE(4,80,PACK(18,K1)) IVAL
0047 FORMAT(A4)
0048 PACK(17,K1)=ICND

```

FORTAN I

0001
0002
0003
0004
0005
0006

0007
0009
0010
0011
0012

FORTRAN IV
0001
0002
0003
0004
0005
0006
0007
0008
0009
0010
0012
0013
0014
0015
0016
0018
0019
0020
0021
0022
0023
0024
0025
0026

PAGE 002

V02.1-1 Mon 04-Jun-79 18:11:55

FORTRAN IV

```
0056 ENCODE(2,90,PACK(22,K1)) MON
0057 FORMAT(A2)
0058 PACK(24,K1)=ITERM
0059 CALL ENSTR(PACK(254,K1),24)
0060 CALL ENABLE(0)
0061 CALL ENQUE(PXING,XING,K1)
0062 CALL ENABLE(1)
0063 RETURN
0064 END
```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 18:12:13      PAGE 001

0001      SUBROUTINE VSQCLP
0002      LOGICAL*1 IDATA
0003      REAL*4 RH,RL,AL,AL,RHT,RLT,ALT,ALT,DELTA,IRSV,VMEAS
0004      REAL*8 VTR
0005      COMMON /VCOM/ ISWITCH,VMEAS(6),IRSV(6),ICTR,RL(6),RH(6),ICHNSW,
1      AL(6),AH(6),DELTA(6),ITERM,NFL,SFL,VTR,IVC,MON,IDATA(10),
2      ICHAN
0006      IF(IDATA(1).EQ.'117'.AND.
1      IDATA(2).EQ.'116') GOTO 50
0008      IF(IDATA(1).EQ.'117'.AND.
1      IDATA(2).EQ.'106'.AND.
2      IDATA(3).EQ.'106') GOTO 60
0010      IF(IDATA(1).EQ.'115') GOTO 70
0012      GOTO 80
0013      ISWITCH=1
0014      ITERM=IDATA(4)
0015      RETURN
0016      ISWITCH=0
0017      RETURN
0018      70      DECODE(4,75,IDATA(2),ERR=80) ICHAN
0019      75      FORMAT(I)
0020      CALL SOUT(ICHAN)
0021      ICHNSW=1
0022      RETURN
0023      STOP
0024      END

```

FORTRAN
 0001
 0002
 0003
 0004
 0005
 0006
 0007
 0008
 0009
 0010
 0011
 0012
 0013
 0014
 0015
 0016
 0017
 0018

```

1  .TITLE FDM.MACRO
2  .SBTTL NODE22
3  .IDENT /V3.0/
4  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WTOKEN,RAM,STATUS,STATRO
6  .GLOBL LPOUT,RSTART,RDFNT,DESTR,ENSTR,LFINPT
7  .NLIST CND
8  .PSECT
9
10 000000
11
12 000001
13 000002
14 000003
15 000004
16 000005
17 000006
18 000007
19
20
21
22
23 172410
24 172412
25 172414
26 172416
27 172416
28 177560
29 177562
30 177564
31 177566
32
33
34
35
36
37
38
39
40
41
42
43 000001
44 000000
45 000001
46 000000
47 000001
48 000000
49 000001
50
51
52 000002
53 000004
54 000006
55 000010
56
57 100000
58 040000
59 000400
60 000200
61 000100
62 000040
63 000020
64 000010
65 000004

```

FORTRAN I

0001
0002
0003
0004
0005
0006
0007
0008
0009
0010
0011
0012
0013
0014
0015
0016
0017
0018
0019
0021
0022
0023
0024
0025
0026
0027
0028
0029
0030
0031
0032
0034
0035
0036
0038
0039
0041
0042
0043
0044
0045
0046
0047
0048
0049
0050
0051
0052
0053

FIM.MACRO
NODE22

46
47
48

MACRO V03.01 4-JUN-79 17:30:32 PAGE 1-1

000002
000001

BIT01= 2
BIT00= 1

FORTTRAN

0054
0055
0056
0057
0058
0059
0060
0061
0062
0063
0064
0065
0066
0067
0068
0069
0070
0071
0072
0073
0074
0075
0076
0077
0078
0079
0080
0081
0082
0083
0084
0085
0086
0087
0088
0089
0090
0091
0092
0094
0095
0096
0097
0098
0099
0101
0102
0103

```

*****
**** PROCEDURE MASTER START UP
*****
MASTER:  MOV    $340, R0
          MTPS   R0
          MOV    $124, R0
          MOV    $110, (R0)+
          MOV    $340, (R0)
          RESET
          MOV    $60, R0
          MOV    $510, (R0)+
          MOV    $340, (R0)+
          MOV    $100, R0
          MOV    $TIME, (R0)
          BIC    $40100, @CSR
          JSR    PC, LIUINT
          CLR    @CSR
          MOV    $000, R0
          MTPS   R0
          RTS    PC

          000000 012700 000340
          7 000004 106400
          8 000006 012700 000124
          9 000012 012720 000430
          10 000016 012710 000340
          18 000022 000005
          20 000024 012700 000060
          25 000030 012720 002470
          26 000034 012720 000340
          34 000040 012700 000100
          35 000044 012710 001654
          36 000050 042737 040100 172414
          37 000056 004767 000014
          38 000062 005037 172414
          39 000066 012700 000000
          40 000072 106400
          41 000074 000207
          66
          67
          68
          69
          70
          71
          72
          73
          74
          75
          76 000076 012737 010402 172416
          77 000104 012737 004400 172416
          78 000112 105737 172414
          79 000116 100375
          80 000120 105037 172414
          81 000124 012737 010401 172416
          82 000132 012700 177400
          83 000136 012737 004407 172416
          84 000144 105737 172414
          85 000150 100375
          86 000152 105037 172414
          87 000156 005200
          88 000160 001366
          90
          91
          92 000162 012701 010410
          93 000166 012700 010440
          94 000172 004767 000106
          95 000176 012737 001400 172416
          96 000204 105737 172414
          97 000210 100375
          98 000212 105037 172414
          99 000216 012701 010610
          100 000222 012700 010640
          101 000226 004767 000052

          ;PRI=7
          ;LIU HANDLER
          ;PRI=7
          ;RESET BUS
          ;SIG INTERFACE HANDLER
          ;CLOCK VECTOR
          ;CLOCK
          ;DISABLE INTERRUPTS
          ;CLEAR LIU
          ;CLEAR BLIUI CSR
          ;PRI=0
          ;CPU DOWN

          ;LDADR
          ;ADDRESS=0
          ;GOOD WD
          ;NO RETRY
          ;SEL ACRAM
          ;COUNTER
          ;WRITE A NULL
          ;GOOD WD
          ;NO LOOP UNTIL READY
          ;COUNT+1

          ;DRUFADR CMD
          ;SEL INBUF0
          ;SET POINTER=0
          ;FALSE READ DATA
          ;GOOD READ
          ;NO RETRY
          ;CLEAR DONE BIT
          ;DRUFADR CMD
          ;SEL INBUF1 COMMAND
          ;POINTER=0

```

102	000232	012737	001400	172416	MOV	#1400, @#OPREG	IFALSE READ DATA
103	000240	105737	172414		TSTB	@#CSR	IGOOD RD
104	000244	100375			BFL	.-4	INO RETRY
105	000246	105037	172414		CLR	@#CSR	ICLEAR DONE BIT
106	000252	012701	010510		MOV	#10510, R1	IRDBUFADR COMMAND
107	000256	012700	010540		MOV	#10540, R0	ISEL OUTBUFO COMMAND
108	000262	004767	000016		JSR	PC, ZEROBF	
109	000266	012701	010710		MOV	#10710, R1	IRDBUFADR COMMAND
110	000272	012700	010740		MOV	#10740, R0	ISEL OUTBUFO COMMAND
111	000276	004767	000002		JSR	PC, ZEROBF	IZERO BUFFER POINTER
112	000302	000440			BR	STAT\$	IGO CLEAR STATUS
113	000304	010137	172416		ZEROBF: MOV	R1, @#OPREG	IRDBUFADR
114	000310	012737	001400	172416	MOV	#1400, @#OPREG	IRD
115	000316	105737	172414		TSTB	@#CSR	IGOOD RD
116	000322	100375			BFL	.-4	INO LOOP UNTIL READY
117	000324	105037	172414		CLR	@#CSR	ICLEAR DONE BIT
118	000330	013702	172416		MOV	@#IDBUF, R2	IFETCH POINTER
119	000334	042702	177400		RIC	#177400, R2	ICLEAR MST BYTE
120	000340	010037	172416		MOV	R0, @#OPREG	ISEL BUFFER
121	000344	022702	000000		CMP	#0, R2	IFINTER=0?
122	000350	001412			BEQ	3\$	IFYES RECHECK
123	000352	012737	001400	172416	MOV	#1400, @#OPREG	INO FALSE RD
124	000360	105737	172414		TSTB	@#CSR	IGOOD RD
125	000364	100375			BFL	.-4	INO LOOP UNTIL READY
126	000366	105037	172414		CLR	@#CSR	ICLEAR DONE BIT
127	000372	005302			DEC	R2	LOCAL POINTER-1
128	000374	000763			BR	2\$	IBR UNTIL 0
129	000376	010037	172416		MOV	R0, @#OPREG	ISEL BUFFER
130	000402	000207			RTS	PC	IRETURN
131							
132							
133							
134	000404	012737	010400	172416	STAT\$: MOV	#4352., @#OPREG	IFWCR:RS
135	000412	012737	002400	172416	MOV	#1280., @#OPREG	IFRS(FALSE)
136	000420	012737	002400	172416	MOV	#1280., @#OPREG	IFRS
137	000426	000207			RTS	PC	
138							
139							

```
1 *****
2 *****
3 *****
4 *****
5 *****
6 000430 010046      MOV      R0,      -(SP)
7 000432 010146      MOV      R1,      -(SP)
8 000434 010246      MOV      R2,      -(SP)
9 000436 010346      MOV      R3,      -(SP)
10 000440 010446      MOV      R4,      -(SP)
11 000442 010546      MOV      R5,      -(SP)
12 000444 005737 172414 TST      @#CSR
13 000450 100153      BFL      RTI$
14 000452 042737      RLC      BIT14, @#CSR
15 000460 012737      MOV      @4352., @#DPREG
16 000466 012737      MOV      @1280., @#DPREG
17 000474 013767      MOV      @#IOBUF, CAUSE
18 000502 042767 177400 BIC      #177400, CAUSE
19
20
21
22 000510 132767 000004      BITB    #BIT02, CAUSE
23 000516 001406      BEQ      B$
24 000520 012702      MOV      @4360., R2
25 000524 012703      MOV      @4384., R3
26 000530 04767 000270      JSR      PC,      EMRF
27 000534 132767 000010      BITB    #BIT03, CAUSE
28 000542 001406      BEQ      B$
29 000544 012702      MOV      @4488., R2
30 000550 012703      MOV      @4512., R3
31 000554 04767 000244      JSR      PC,      EMRF
32 000560 132767 000020      BITB    #BIT04, CAUSE
33 000566 001407      BEQ      B$
34 000570 132767 000200      BITB    #BIT07, CAUSE
35 000576 001403      BEQ      B$
36 000600 012767 000002      MOV      @2,      LLFLG
37 000606 132767 000040      BITB    #BIT05, CAUSE
38 000614 001403      BEQ      B$
39 000616 012767 000001      MOV      @1,      LLFLG
40 000624 132767 000100      BITB    #BIT06, CAUSE
41 000632 001404      BEQ      B$
42 000634 005067      CLR      IWRTH
43 000640 005067      CLR      IWRTH
44
45
46
47 000644 132767 000001      BITB    #BIT00, CAUSE
48 000652 001423      BEQ      B1$
49 000654 012701      MOV      @INEF, R1
50 000660 012702      MOV      @4360., R2
51 000664 012703      MOV      @4384., R3
52 000670 012704      MOV      @BIT00, R4
53 000674 004767 000124      JSR      PC,      EMRF
54 000700 012705 001206      MOV      @AREA, R5
55 000704 012767 000001      MOV      @1,      AREA
56 000712 010267 000274      MOV      R2,      DATA
57 000716 004767 000000      JSR      PC,      LPINFT

*****
***** PROCEDURE LIU-HANDLER *****
*****
*****
;SAVE R0
;SAVE R1
;SAVE R2
;SAVE R3
;SAVE R4
;SAVE R5
;? LIU INTERRUPTED
;NO
;YES/DISABLE LIU
;WCR : RS(0)
;RD
;CLEAR UNUSED BITS

;OV-FL
;RDBUFADR INO
;SEL INBUFO
;GO EMPTY
;OV-FL
;RDBUFADR IN1
;SEL INRUF1
;GO EMPTY
;LINE-LOSS PRIMARY
;LATCH ON
;YES LOOP AROUND
;FLAG PRIMARY SW. FAIL
;BACKUP LINE LOSS
;FLAG BACKUP SW. FAIL
;WRITE TOKEN DETECT
;CLEAR TOKEN VAR

;INBUFO FULL
;NO
;RDBUFADR ADDRESS
;RDBUFADR COMM
;SEL RUF
;CRC BIT
;EMPTY BUFFER
;DATA LINK AREA
;I VARIABLE
;CRC OR BYTE COUNT
;CALL FORTRAN QUE'ER
```



```

58 000722 132767 000002 000254 R1$: R1T01, CAUSE
59 000730 001423 BEQ RTI$
60 000732 012701 MOV #INBF, R1
61 000736 012702 MOV #4488., R2
62 000742 012703 MOV #4512., R3
63 000746 012704 MOV #R1T01, R4
64 000752 004767 JSR PC, EMBF
65 000756 012705 MOV #AREA, R5
66 000762 012767 MOV #1, AREA
67 000770 010267 MOV R2, DATA
68 000774 004767 JSR PC, LFINFT
69
70 001000 012605 RTI$: MOV (SP)+, R5
71 001002 012604 MOV (SP)+, R4
72 001004 012603 MOV (SP)+, R3
73 001006 012602 MOV (SP)+, R2
74 001010 012601 MOV (SP)+, R1
75 001012 012600 MOV (SP)+, R0
76 001014 052737 BIS #BIT14, @%CSR
77 001022 000002 RTI
78
79 001024 012737 010600 172416 EMBF: MOV #4480., @%DFREG
80 001032 012737 002400 172416 MOV #1280., @%DFREG
81 001040 013700 MOV @%IOBUF, R0
82 001044 130400 BTR R4,R0
83 001046 001002 BNE CRCOK
84 001050 012704 MOV #1, R4
85 001054 010237 172416 MOV R2, @%DFREG
86 001060 012737 001400 172416 MOV #768., @%DFREG
87 001066 105737 172414 TSTB @%CSR
88 001072 100375 BPL .-4
89 001074 013702 172416 MOV @%IOBUF, R2
90 001100 042702 177400 BIC #177400, R2
91 001104 010200 MOV R0, R0
92 001106 005400 NEG R0
93 001110 010037 172412 MOV R0, @%WCR
94 001114 010137 172410 MOV R1, @%BAR
95 001120 010337 172416 MOV R3, @%DFREG
96 001124 012737 001400 172416 MOV #768., @%DFREG
97 001132 105737 172414 TSTB @%CSR
98 001136 100375 BPL .-4
99 001140 012737 021000 172416 MOV #8704., @%DFREG
100 001146 000240 NOP
101 001150 105737 172414 TSTB @%CSR
102 001154 100401 BMI DMAOK
103 001156 000240 NOP
104 001160 012737 004400 172416 DMAOK: MOV #2304., @%DFREG
105 001166 105737 172414 TSTB @%CSR
106 001172 100375 BPL .-4
107 001174 005704 TST R4
108 001176 100001 BPL END$
109 001200 010402 MOV R4, R2
110 001202 000207 RTS FC
111 001204 000000 .WORD 0
112 001206 000000 .WORD 0
113 001210 001212 .WORD DATA
114 001212 000000 .WORD 0

```

```

;INBUF1 FULL
;NO WERE DONE
;BUFFER ADDRESS
;RDBUFADR COMM
;SEL INBUF1
;CRC BIT
;GO EMPTY BUFFER
;DATA LINK AREA
;ONE VARIABLE
;CRC OR BYTE COUNT
;CALL FORTRAN QUE'ER
;RESTORE REGISTERS

;ENABLE INTERRUPTS
;RETURN FROM INTERRUPT

;READ STATUS 1
;RS
;FETCH STATUS
;GOOD CRC

;NO FLAG
;RDBUFADR
;RD
;GOOD RD

;REPLACE WITH POINTER
;CLEAR MST BITS
;SAVE IT
;2'S COMP
;BYTE COUNT NOW
;ADDRESS IN MEMORY
;SEL BUFFER
;FALSE RD
;DONE ON
;NO LOOP
;FIRE DMA
;DELAY
;GOOD DMA

;ERROR IF HERE
;FALSE WD
;GOOD WD

;WAS CRC OK
;YES
;NO FLAG IT
;RETURN
;STATUS BYTE 0 HOLDER

```


[illegible]

```
1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****

;CALL STATUS(X,DATA) -READ STATUS BYTES 0/1
;
;X=0 STATUS BYTE 0
;X=1 STATUS BYTE 1

STATUS: MOV @ARG1(R5), R1
11 CMP #1, R1
12 BEQ 2$
13 MOV #4352., @OPREG
14 MOV #1280., @OPREG
15 MOV @IOBUF, R0
16 BIC #177400, R0
17 MOV R0, @ARG2(R5)
18 RTS
19 MOV #4480., @OPREG
20 MOV #1280., @OPREG
21 MOV @IOBUF, R0
22 BIC #177400, R0
23 MOV R0, @ARG2(R5)
24 RTS
25 MOV #4352., @OPREG
26 MOV #1280., @OPREG
27 MOV @IOBUF, R1
28 BIC #177400, R1
29 BITB #BIT03, R1
30 BEQ 3$
31 MOV #1, @ARG1(R5)
32 RTS
33 MOV #0, @ARG1(R5)
34 RTS
35
36

;WHICH BYTE
;COMPARE
;WCR : RS(0)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;WCR : RS(1)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;STATUS BYTE 1
;RS
;FETCH DATA
;BUFFER FULL
;LOAD RETURN
;LOAD RETURN
```

```

1 *****
2 ***** PROCEDURE WRITE TOKEN (WTKEN) *****
3 *****
4 *****
5 *****
6 *****
7 001474 012737 010600 172416 WTKEN: MOV $4480., @#DPREG
8 001502 012737 002400 172416 MOV $1280., @#DPREG
9 001510 013701 172416 MOV @#I0BUF, R1
10 001514 132701 000010 BITB @#BIT03, R1
11 001520 001414 REQ WTKG
12 001522 012737 010420 172416 MOV $4368., @#DPREG
13 001530 012737 004423 172416 MOV $2323., @#DPREG
14 001536 105737 172414 TSTB @#CSR
15 001542 100375 RPL -4
16 001544 105037 172414 CLRB @#CSR
17 001550 000207 RTS FC
18 001552 012737 010740 172416 WTKG: MOV $4576., @#DPREG
19 001560 012737 001652' 172410 WTD, @#BAR
20 001566 012702 000002 MOV $2, R2
21 001572 005402 NEG R2
22 001574 010237 172412 MOV R2, @#WCR
23 001600 012737 024000 172416 MOV $10240., @#DPREG
24 001606 000240 NOP
25 001610 105737 172414 TSTB @#CSR
26 001614 000240 NOP
27 001616 105037 172414 CLRB @#CSR
28 001622 012737 010420 172416 MOV $4368., @#DPREG
29 001630 012737 004422 172416 MOV $2322., @#DPREG
30 001636 105737 172414 TSTB @#CSR
31 001642 100375 RPL -4
32 001644 105037 172414 CLRB @#CSR
33 001650 000207 RTS FC
34 001652 000 377 WTD: .BYTE 0,377
35
36
37 *****
38 ***** PROCEDURE TIMER (TIME) *****
39 *****
40 *****
41 001654 005267 001006' TIME: INC LTIME
42 001660 022767 077777 001006' CMP $077777, LTIME
43 001666 001002 BNE 4$
44 001670 005067 001006' CLR LTIME
45 001674 000002 RTI
46
47
48

```

```
1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****

;CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;0100 =DREAD
;0111 =NULL
;0000 =WTKEN

;*****
;**** PROCEDURE ACRAM
;*****

;RAM
;WHICH OPERATION
;READ OP
;ADDRESS
;WRITE DATA
;SEL LDACR
;WD/DATA (ADDR)
;WRITE DATA
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;SEL ACRAM
;WD/DATA (CMD)
;WRITE
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT

;ADDRESS TO READ
;SEL LDACR
;WD/DATA (ADDR)
;WRITE DATA
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;SEL ACRAM
;READ DATA
;VALID READ
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;FEICH DATA
;CLEAR BITS
;RIN DATA
```



```

1      .ENABLE LSB
2      *****
3      ***** PROCEDURE WRITE LOOP (LPOUT) *****
4      *****
5
6      002102 017502 000002 LPOUT: MOV @ARG1(R5), R2
7      002106 012700 000340 MOV $340, R0
8      002112 106400 MTF5 R0
9      002114 012737 000000' 172410 MOV $OUTBF, @BBAR
10     002122 005402 NEG R2
11     002124 010237 172412 MOV R2, @WCR
12     002130 012737 010540 MOV $4448, @OPREG
13     002136 012737 024000 172416 MOV $10240, @OPREG
14     002144 000240 NOP @CSR
15     002146 105737 172414 TSTB @CSR
16     002152 000240 NOP @CSR
17     002154 105037 172414 MOV $4576, @OPREG
18     002160 012737 010740 172416 MOV $WTD, @BBAR
19     002166 012737 001652' 172410 MOV $2, R2
20     002174 012702 000002 NEG R2
21     002200 005402 MOV R2, @WCR
22     002202 010237 172412 MOV $10240, @OPREG
23     002206 012737 024000 172416 MOV @CSR
24     002214 000240 NOP @CSR
25     002216 105737 172414 TSTB @CSR
26     002222 000240 NOP @CSR
27     002224 105037 172414 MOV $4368, @OPREG
28     002230 012737 010420 172416 MOV $2307, @OPREG
29     002236 012737 004403 172416 TSTB @CSR
30     002244 105737 172414 BPL -4
31     002250 100375 @CSR
32     002252 105037 172414 MOV $000, R0
33     002256 012700 000000 MTF5 R0
34     002262 106400 RTS
35     002264 000207
36
37
38

```



```

1 1 5 002266 017500 000002
2 2 6 002272 022700 000000
3 3 7 002276 001004
4 8 002300 012700 010410
5 9 002304 000167 000046
6 10 002310 022700 000001
7 11 002314 001004
8 12 002316 012700 010610
9 13 002322 000167 000030
10 14 002326 022700 000002
11 15 002332 001004
12 16 002334 012700 010510
13 17 002340 000167 000012
14 18 002344 022700 000004
15 19 002350 001020
16 20 002352 012700 010710
17 21 002356 010037 172416
18 22 002362 012737 001400
19 23 002370 105737 172414
20 24 002374 100375
21 25 002376 013700 172416
22 26 002402 042700 177400
23 27 002406 010075 000004
24 28 002412 000207

RDINT: MOV @ARG1(R5), R0
        CMP #0, R0
        BNE 1$
        MOV #4360., R0
        JMF 4$
        CMP #1, R0
        BNE 2$
        MOV #4488., R0
        JMF 4$
        CMP #2, R0
        BNE 3$
        MOV #4424., R0
        JMF 4$
        CMP #4, R0
        BNE 5$
        MOV #4552., R0
        JMF 4$
        MOV R0, @OPREG
        MOV #768., @OPREG
        TSTB @CSR
        BPL -4
        MOV @#IOBUF, R0
        BIC #177400, R0
        MOV R0, @ARG2(R5)
        RTS

1$:
2$:
3$:
4$:
5$:

;FEICH COMMAND
;INBUFO ?
;NO
;RDBUFADR IN0
;INBUF1 ?
;NO
;RDBUFADR IN1
;OUTBUFO ?
;NO
;RDBUFADR OUT0
;OUTBUF1 ?
;NO RETURN
;RDBUFADR OUT1
;RDBUFADR
;RD
;READY
;NO LOOP UNTIL
;FEICH POINTER
;CLEAR MST
;RETURN POINTER

```

```

1  *****
2  *****
3  *****
4  *****
5  *****
6  *****
7  *****
8  *****
9  002414 017500 000002 000000 000000
10 002420 022700 000000 000000 000000
11 002424 001002
12 002426 000000
13 002430 000207
14 002432 022700 000001
15 002436 001005
16 002440 004767 175432
17 002444 012700 000040
18 002450 000110
19 002452 022700 000002
20 002456 001002
21 002460 000177 000002
22 002464 000207
23 002466 173000
24
25
26

;CALL RSTART(0) -CAUSES A SOFTWARE HALT
;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
;CALL RSTART(2) -LOAD MODE(173000)

*****
;*** PROCEDURE RESTART
*****
;
;
;MODE
;LOAD ADDRESS
;
;RESTART PROGRAM
;INT LIU FIRST
;START ADDRESS
;LOAD MODE

RSTART: MOV @ARG1(RS), R0
RS0: CMP #0, R0
BNE RS1
HALT
RTS PC
RS1: CMP #1, R0
BNE RS2
JSR PC, LIUINT
G4: MOV #40, R0
JMP (R0)
RS2: CMP #2, R0
BNE RST
JMP @ROM
RST: RTS PC
ROM: .WORD 173000

```

```

1      .ENABLE LSB
2      *****
3      ***** PROCEDURE SIG INTERFACE *****
4      *****
5
6      002470 010046      SIO::      MOV      R0,      -(SP)
7      002472 010146      MOV      R1,      -(SP)
8      002474 010246      MOV      R2,      -(SP)
9      002476 012700      MOV      R2,      XINBF, R0
10     002502 012701      MOV      R0,      XINBF, R0
11     002506 000400'     MOV      R0,      XINBF, R0
12     002512 000050      MOV      R0,      XINBF, R0
13     002514 000050      MOV      R0,      XINBF, R0
14     002520 177560      TSTB     @#HRCR
15     002524 177562      BPL      -4
16     002530 177562      MOV      R0,      XINBF, R0
17     002532 177562      BIC      R2,      IOFLG
18     002534 177562      MOV      R2,      IOFLG
19     002540 177562      DEC      R1
20     002542 177562      TSTB     @#HRCR
21     002546 177562      BPL      -4
22     002552 177562      MOV      R0,      XINBF, R0
23     002554 177562      BIC      R2,      IOFLG
24     002556 177562      MOV      R2,      IOFLG
25     002560 177562      DEC      R1
26     002564 177562      TSTB     @#HRCR
27     002570 177562      BPL      -4
28     002572 177562      MOV      R0,      XINBF, R0
29     002574 177562      BIC      R2,      IOFLG
30     002576 177562      MOV      R2,      IOFLG
31     002578 177562      DEC      R1
32     002580 177562      TSTB     @#HRCR
33     002582 177562      BPL      -4
34     002584 177562      MOV      R0,      XINBF, R0
35     002586 177562      BIC      R2,      IOFLG
36     002588 177562      MOV      R2,      IOFLG
37     002590 177562      DEC      R1
38     002592 177562      TSTB     @#HRCR
39     002594 177562      BPL      -4
40     002596 177562      MOV      R0,      XINBF, R0
41     002598 177562      BIC      R2,      IOFLG
42     002600 177562      MOV      R2,      IOFLG
43     002602 177562      DEC      R1
44     002604 177562      TSTB     @#HRCR
45     002606 177562      BPL      -4
46     002608 177562      MOV      R0,      XINBF, R0
47     002610 177562      BIC      R2,      IOFLG
48     002612 177562      MOV      R2,      IOFLG
49     002614 177562      DEC      R1
50     002616 177562      TSTB     @#HRCR
51     002618 177562      BPL      -4
52     002620 177562      MOV      R0,      XINBF, R0
53     002622 177562      BIC      R2,      IOFLG
54     002624 177562      MOV      R2,      IOFLG
55     002626 177562      DEC      R1
56     002628 177562      TSTB     @#HRCR
57     002630 177562      BPL      -4
58     002632 177562      MOV      R0,      XINBF, R0
59     002634 177562      BIC      R2,      IOFLG
60     002636 177562      MOV      R2,      IOFLG
61     002638 177562      DEC      R1
62     002640 177562      TSTB     @#HRCR
63     002642 177562      BPL      -4
64     002644 177562      MOV      R0,      XINBF, R0
65     002646 177562      BIC      R2,      IOFLG
66     002648 177562      MOV      R2,      IOFLG
67     002650 177562      DEC      R1
68     002652 177562      TSTB     @#HRCR
69     002654 177562      BPL      -4
70     002656 177562      MOV      R0,      XINBF, R0
71     002658 177562      BIC      R2,      IOFLG
72     002660 177562      MOV      R2,      IOFLG
73     002662 177562      DEC      R1
74     002664 177562      TSTB     @#HRCR
75     002666 177562      BPL      -4
76     002668 177562      MOV      R0,      XINBF, R0
77     002670 177562      BIC      R2,      IOFLG
78     002672 177562      MOV      R2,      IOFLG
79     002674 177562      DEC      R1
80     002676 177562      TSTB     @#HRCR
81     002678 177562      BPL      -4
82     002680 177562      MOV      R0,      XINBF, R0
83     002682 177562      BIC      R2,      IOFLG
84     002684 177562      MOV      R2,      IOFLG
85     002686 177562      DEC      R1
86     002688 177562      TSTB     @#HRCR
87     002690 177562      BPL      -4
88     002692 177562      MOV      R0,      XINBF, R0
89     002694 177562      BIC      R2,      IOFLG
90     002696 177562      MOV      R2,      IOFLG
91     002698 177562      DEC      R1
92     002700 177562      TSTB     @#HRCR
93     002702 177562      BPL      -4
94     002704 177562      MOV      R0,      XINBF, R0
95     002706 177562      BIC      R2,      IOFLG
96     002708 177562      MOV      R2,      IOFLG
97     002710 177562      DEC      R1
98     002712 177562      TSTB     @#HRCR
99     002714 177562      BPL      -4
100    002716 177562      MOV      R0,      XINBF, R0
101    002718 177562      BIC      R2,      IOFLG
102    002720 177562      MOV      R2,      IOFLG
103    002722 177562      DEC      R1
104    002724 177562      TSTB     @#HRCR
105    002726 177562      BPL      -4
106    002728 177562      MOV      R0,      XINBF, R0
107    002730 177562      BIC      R2,      IOFLG
108    002732 177562      MOV      R2,      IOFLG
109    002734 177562      DEC      R1
110    002736 177562      TSTB     @#HRCR
111    002738 177562      BPL      -4
112    002740 177562      MOV      R0,      XINBF, R0
113    002742 177562      BIC      R2,      IOFLG
114    002744 177562      MOV      R2,      IOFLG
115    002746 177562      DEC      R1
116    002748 177562      TSTB     @#HRCR
117    002750 177562      BPL      -4
118    002752 177562      MOV      R0,      XINBF, R0
119    002754 177562      BIC      R2,      IOFLG
120    002756 177562      MOV      R2,      IOFLG
121    002758 177562      DEC      R1
122    002760 177562      TSTB     @#HRCR
123    002762 177562      BPL      -4
124    002764 177562      MOV      R0,      XINBF, R0
125    002766 177562      BIC      R2,      IOFLG
126    002768 177562      MOV      R2,      IOFLG
127    002770 177562      DEC      R1
128    002772 177562      TSTB     @#HRCR
129    002774 177562      BPL      -4
130    002776 177562      MOV      R0,      XINBF, R0
131    002778 177562      BIC      R2,      IOFLG
132    002780 177562      MOV      R2,      IOFLG
133    002782 177562      DEC      R1
134    002784 177562      TSTB     @#HRCR
135    002786 177562      BPL      -4
136    002788 177562      MOV      R0,      XINBF, R0
137    002790 177562      BIC      R2,      IOFLG
138    002792 177562      MOV      R2,      IOFLG
139    002794 177562      DEC      R1
140    002796 177562      TSTB     @#HRCR
141    002798 177562      BPL      -4
142    002800 177562      MOV      R0,      XINBF, R0
143    002802 177562      BIC      R2,      IOFLG
144    002804 177562      MOV      R2,      IOFLG
145    002806 177562      DEC      R1
146    002808 177562      TSTB     @#HRCR
147    002810 177562      BPL      -4
148    002812 177562      MOV      R0,      XINBF, R0
149    002814 177562      BIC      R2,      IOFLG
150    002816 177562      MOV      R2,      IOFLG
151    002818 177562      DEC      R1
152    002820 177562      TSTB     @#HRCR
153    002822 177562      BPL      -4
154    002824 177562      MOV      R0,      XINBF, R0
155    002826 177562      BIC      R2,      IOFLG
156    002828 177562      MOV      R2,      IOFLG
157    002830 177562      DEC      R1
158    002832 177562      TSTB     @#HRCR
159    002834 177562      BPL      -4
160    002836 177562      MOV      R0,      XINBF, R0
161    002838 177562      BIC      R2,      IOFLG
162    002840 177562      MOV      R2,      IOFLG
163    002842 177562      DEC      R1
164    002844 177562      TSTB     @#HRCR
165    002846 177562      BPL      -4
166    002848 177562      MOV      R0,      XINBF, R0
167    002850 177562      BIC      R2,      IOFLG
168    002852 177562      MOV      R2,      IOFLG
169    002854 177562      DEC      R1
170    002856 177562      TSTB     @#HRCR
171    002858 177562      BPL      -4
172    002860 177562      MOV      R0,      XINBF, R0
173    002862 177562      BIC      R2,      IOFLG
174    002864 177562      MOV      R2,      IOFLG
175    002866 177562      DEC      R1
176    002868 177562      TSTB     @#HRCR
177    002870 177562      BPL      -4
178    002872 177562      MOV      R0,      XINBF, R0
179    002874 177562      BIC      R2,      IOFLG
180    002876 177562      MOV      R2,      IOFLG
181    002878 177562      DEC      R1
182    002880 177562      TSTB     @#HRCR
183    002882 177562      BPL      -4
184    002884 177562      MOV      R0,      XINBF, R0
185    002886 177562      BIC      R2,      IOFLG
186    002888 177562      MOV      R2,      IOFLG
187    002890 177562      DEC      R1
188    002892 177562      TSTB     @#HRCR
189    002894 177562      BPL      -4
190    002896 177562      MOV      R0,      XINBF, R0
191    002898 177562      BIC      R2,      IOFLG
192    002900 177562      MOV      R2,      IOFLG
193    002902 177562      DEC      R1
194    002904 177562      TSTB     @#HRCR
195    002906 177562      BPL      -4
196    002908 177562      MOV      R0,      XINBF, R0
197    002910 177562      BIC      R2,      IOFLG
198    002912 177562      MOV      R2,      IOFLG
199    002914 177562      DEC      R1
200    002916 177562      TSTB     @#HRCR
201    002918 177562      BPL      -4
202    002920 177562      MOV      R0,      XINBF, R0
203    002922 177562      BIC      R2,      IOFLG
204    002924 177562      MOV      R2,      IOFLG
205    002926 177562      DEC      R1
206    002928 177562      TSTB     @#HRCR
207    002930 177562      BPL      -4
208    002932 177562      MOV      R0,      XINBF, R0
209    002934 177562      BIC      R2,      IOFLG
210    002936 177562      MOV      R2,      IOFLG
211    002938 177562      DEC      R1
212    002940 177562      TSTB     @#HRCR
213    002942 177562      BPL      -4
214    002944 177562      MOV      R0,      XINBF, R0
215    002946 177562      BIC      R2,      IOFLG
216    002948 177562      MOV      R2,      IOFLG
217    002950 177562      DEC      R1
218    002952 177562      TSTB     @#HRCR
219    002954 177562      BPL      -4
220    002956 177562      MOV      R0,      XINBF, R0
221    002958 177562      BIC      R2,      IOFLG
222    002960 177562      MOV      R2,      IOFLG
223    002962 177562      DEC      R1
224    002964 177562      TSTB     @#HRCR
225    002966 177562      BPL      -4
226    002968 177562      MOV      R0,      XINBF, R0
227    002970 177562      BIC      R2,      IOFLG
228    002972 177562      MOV      R2,      IOFLG
229    002974 177562      DEC      R1
230    002976 177562      TSTB     @#HRCR
231    002978 177562      BPL      -4
232    002980 177562      MOV      R0,      XINBF, R0
233    002982 177562      BIC      R2,      IOFLG
234    002984 177562      MOV      R2,      IOFLG
235    002986 177562      DEC      R1
236    002988 177562      TSTB     @#HRCR
237    002990 177562      BPL      -4
238    002992 177562      MOV      R0,      XINBF, R0
239    002994 177562      BIC      R2,      IOFLG
240    002996 177562      MOV      R2,      IOFLG
241    002998 177562      DEC      R1
242    003000 177562      TSTB     @#HRCR
243    003002 177562      BPL      -4
244    003004 177562      MOV      R0,      XINBF, R0
245    003006 177562      BIC      R2,      IOFLG
246    003008 177562      MOV      R2,      IOFLG
247    003010 177562      DEC      R1
248    003012 177562      TSTB     @#HRCR
249    003014 177562      BPL      -4
250    003016 177562      MOV      R0,      XINBF, R0
251    003018 177562      BIC      R2,      IOFLG
252    003020 177562      MOV      R2,      IOFLG
253    003022 177562      DEC      R1
254    003024 177562      TSTB     @#HRCR
255    003026 177562      BPL      -4
256    003028 177562      MOV      R0,      XINBF, R0
257    003030 177562      BIC      R2,      IOFLG
258    003032 177562      MOV      R2,      IOFLG
259    003034 177562      DEC      R1
260    003036 177562      TSTB     @#HRCR
261    003038 177562      BPL      -4
262    003040 177562      MOV      R0,      XINBF, R0
263    003042 177562      BIC      R2,      IOFLG
264    003044 177562      MOV      R2,      IOFLG
265    003046 177562      DEC      R1
266    003048 177562      TSTB     @#HRCR
267    003050 177562      BPL      -4
268    003052 177562      MOV      R0,      XINBF, R0
269    003054 177562      BIC      R2,      IOFLG
270    003056 177562      MOV      R2,      IOFLG
271    003058 177562      DEC      R1
272    003060 177562      TSTB     @#HRCR
273    003062 177562      BPL      -4
274    003064 177562      MOV      R0,      XINBF, R0
275    003066 177562      BIC      R2,      IOFLG
276    003068 177562      MOV      R2,      IOFLG
277    003070 177562      DEC      R1
278    003072 177562      TSTB     @#HRCR
279    003074 177562      BPL      -4
280    003076 177562      MOV      R0,      XINBF, R0
281    003078 177562      BIC      R2,      IOFLG
282    003080 177562      MOV      R2,      IOFLG
283    003082 177562      DEC      R1
284    003084 177562      TSTB     @#HRCR
285    003086 177562      BPL      -4
286    003088 177562      MOV      R0,      XINBF, R0
287    003090 177562      BIC      R2,      IOFLG
288    003092 177562      MOV      R2,      IOFLG
289    003094 177562      DEC      R1
290    003096 177562      TSTB     @#HRCR
291    003098 177562      BPL      -4
292    003100 177562      MOV      R0,      XINBF, R0
293    003102 177562      BIC      R2,      IOFLG
294    003104 177562      MOV      R2,      IOFLG
295    003106 177562      DEC      R1
296    003108 177562      TSTB     @#HRCR
297    003110 177562      BPL      -4
298    003112 177562      MOV      R0,      XINBF, R0
299    003114 177562      BIC      R2,      IOFLG
300    003116 177562      MOV      R2,      IOFLG
301    003118 177562      DEC      R1
302    003120 177562      TSTB     @#HRCR
303    003122 177562      BPL      -4
304    003124 177562      MOV      R0,      XINBF, R0
305    003126 177562      BIC      R2,      IOFLG
306    003128 177562      MOV      R2,      IOFLG
307    003130 177562      DEC      R1
308    003132 177562      TSTB     @#HRCR
309    003134 177562      BPL      -4
310    003136 177562      MOV      R0,      XINBF, R0
311    003138 177562      BIC      R2,      IOFLG
312    003140 177562      MOV      R2,      IOFLG
313    003142 177562      DEC      R1
314    003144 177562      TSTB     @#HRCR
315    003146 177562      BPL      -4
316    003148 177562      MOV      R0,      XINBF, R0
317    003150 177562      BIC      R2,      IOFLG
318    003152 177562      MOV      R2,      IOFLG
319    003154 177562      DEC      R1
320    003156 177562      TSTB     @#HRCR
321    003158 177562      BPL      -4
322    003160 177562      MOV      R0,      XINBF, R0
323    003162 177562      BIC      R2,      IOFLG
324    003164 177562      MOV      R2,      IOFLG
325    003166 177562      DEC      R1
326    003168 177562      TSTB     @#HRCR
327    003170 177562      BPL      -4
328    003172 177562      MOV      R0,      XINBF, R0
329    003174 177562      BIC      R2,      IOFLG
330    003176 177562      MOV      R2,      IOFLG
331    003178 177562      DEC      R1
332    003180 177562      TSTB     @#HRCR
333    003182 177562      BPL      -4
334    003184 177562      MOV      R0,      XINBF, R0
335    003186 177562      BIC      R2,      IOFLG
336    003188 177562      MOV      R2,      IOFLG
337    003190 177562      DEC      R1
338    003192 177562      TSTB     @#HRCR
339    003194 177562      BPL      -4
340    003196 177562      MOV      R0,      XINBF, R0
341    003198 177562      BIC      R2,      IOFLG
342    003200 177562      MOV      R2,      IOFLG
343    003202 177562      DEC      R1
344    003204 177562      TSTB     @#HRCR
345    003206 177562      BPL      -4
346    003208 177562      MOV      R0,      XINBF, R0
347    003210 177562      BIC      R2,      IOFLG
348    003212 177562      MOV      R2,      IOFLG
349    003214 177562      DEC      R1
350    003216 177562      TSTB     @#HRCR
351    003218 177562      BPL      -4
352    003220 177562      MOV      R0,      XINBF, R0
353    003222 177562      BIC      R2,      IOFLG
354    003224 177562      MOV      R2,      IOFLG
355    003226 177562      DEC      R1
356    003228 177562      TSTB     @#HRCR
357    003230 177562      BPL      -4
358    003232 177562      MOV      R0,      XINBF, R0
359    003234 177562      BIC      R2,      IOFLG
360    003236 177562      MOV      R2,      IOFLG
361    003238 177562      DEC      R1
362    003240 177562      TSTB     @#HRCR
363    003242 177562      BPL      -4
364    003244 177562      MOV      R0,      XINBF, R0
365    003246 177562      BIC      R2,      IOFLG
366    003248 177562      MOV      R2,      IOFLG
367    003250 177562      DEC      R1
368    003252 177562      TSTB     @#HRCR
369    003254 177562      BPL      -4
370    003256 177562      MOV      R0,      XINBF, R0
371    003258 177562      BIC      R2,      IOFLG
372    003260 177562      MOV      R2,      IOFLG
373    003262 177562      DEC      R1
374    003264 177562      TSTB     @#HRCR
375    003266 177562      BPL      -4
376    003268 177562      MOV      R0,      XINBF, R0
377    003270 177562      BIC      R2,      IOFLG
378    003272 177562      MOV      R2,      IOFLG
379    003274 177562      DEC      R1
380    003276 177562      TSTB     @#HRCR
381    003278 177562      BPL      -4
382    003280 177562      MOV      R0,      XINBF, R0
383    003282 177562      BIC      R2,      IOFLG
384    003284 177562      MOV      R2,      IOFLG
385    003286 177562      DEC      R1
386    003288 177562      TSTB     @#HRCR
387    003290 177562      BPL      -4
388    003292 177562      MOV      R0,      XINBF, R0
389    003294 177562      BIC      R2,      IOFLG
390    003296 177562      MOV      R2,      IOFLG
391    003298 177562      DEC      R1
392    003300 177562      TSTB     @#HRCR
393    003302 177562      BPL      -4
394    003304 177562      MOV      R0,      XINBF, R0
395    003306 177562      BIC      R2,      IOFLG
396    003308 177562      MOV      R2,      IOFLG
397    003310 177562      DEC      R1
398    003312 177562      TSTB     @#HRCR
399    003314 177562      BPL      -4
400    003316 177562      MOV      R0,      XINBF, R0
401    003318 177562      BIC      R2,      IOFLG
402    003320 177562      MOV      R2,      IOFLG
403    003322 177562      DEC      R1
404    003324 177562      TSTB     @#HRCR
405    003326 177562      BPL      -4
406    003328 177562      MOV      R0,      XINBF, R0
407    003330 177562      BIC      R2,      IOFLG
408    003332 177562      MOV      R2,      IOFLG
409    003334 177562      DEC      R1
410    003336 177562      TSTB     @#HRCR
411    003338 177562      BPL      -4
412    003340 177562      MOV      R0,      XINBF, R0
413    003342 177562      BIC      R2,      IOFLG
414    003344 177562      MOV      R2,      IOFLG
415    003346 177562      DEC      R1
416    003348 177562      TSTB     @#HRCR
417    003350 177562      BPL      -4
418    003352 177562      MOV      R0,      XINBF, R0
419    003354 177562      BIC      R2,      IOFLG
420    003356 177562      MOV      R2,      IOFLG
421    003358 177562      DEC      R1
422    003360 177562      TSTB     @#HRCR
423    003362 177562      BPL      -4
424    003364 177562      MOV      R0,      XINBF, R0
425    003366 177562      BIC      R2,      IOFLG
426    003368 177562      MOV      R2,      IOFLG
427    003370 177562      DEC      R1
428    003372 177562      TSTB     @#HRCR
429    003374 177562      BPL      -4
430    003376 177562      MOV      R0,      XINBF, R0
431    003378 177562      BIC      R2,      IOFLG
432    003380 177562      MOV      R2,      IOFLG
433    003382 177562      DEC      R1
434    003384 177562      TSTB     @#HRCR
435    003386 177562      BPL      -4
436    003388 177562      MOV      R0,      XINBF, R0
437    003390 177562      BIC      R2,      IOFLG
438    003392 177562      MOV      R2,      IOFLG
439    003394 177562      DEC      R1
440    003396 177562      TSTB     @#HRCR
441    003398 177562      BPL      -4
442    003400 177562      MOV      R
```

FDM-MACRO MACRO V03.01 4-JUN-79 17:30:32 PAGE 12
 NODE22

```

1 *****
2 *****
3 ***** COMMON DATA AREA *****
4 *****
5 0000000 .PSECT DFM,RW,D,GBL,REL,OVR
6
7 0000000 OUTBF: .BLKB 256.
8 0004000 INBF: .BLKB 256.
9 0010000 IWRTTM: .BLKW
10 0010002 IWRT: .BLKW
11 0010004 STAT: .BLKW
12 0010006 LTIME: .BLKW
13 .EVEN
14
15 0000000 .PSECT BUFS,RW,D,GBL,REL,OVR
16
17 0000000 XOUTBF: .BLKB 256.
18 0004000 XINBF: .BLKB 256.
19 0010000 ACKSEQ: .BLKB 256.
20 0014000 LIDFD: .BLKB 256.
21 0020000 IOFLG: .BLKW
22 0020002 LLFLG: .BLKW
23 0020004 IRSEND: .BLKW
24 .EVEN
25
26
27 0000001 .END

```

FDM-MACRO MACRO V03.01 4-JUN-79 17:30:32 PAGE 12-1
SYMBOL TABLE

ACKSEQ	001000R	003	B1\$	000722R	E1\$	001236R	MASTER	000000RG	SOUT	002576RG	002
AREA	001206R	E2\$		000510R	G4\$	002444R	NDXX =	000000	STAT	001004R	
ARG1	= 000002	B3\$		000534R	HREUF =	177562	ND24 =	000001	STATO	001422RG	
ARG2	= 000004	B4\$		000560R	HRCUR =	177560	NHOST =	000000	STATUS	001324RG	
ARG3	= 000006	B5\$		000606R	HXBUR =	177566	OPREG =	172416	STAT\$	000404R	
ARG4	= 000010	B6\$		000624R	HXCSR =	177564	OUTBF	000000R	002 SWITCH	001266RG	
BAR	= 172410	B7\$		000570R	INBF	000400R	002 RAM	001676RG	TIME	001654RG	
B1T00	= 000001	CAUSE		001204R	IOBUF =	172416	KDPNT	002266RG	WCR =	172412	
B1T01	= 000002	CRCOK		001054R	IOFLG	002000R	003 RDRAM	002002R	WTD	001652R	
B1T02	= 000004	CSR	= 172414	IRSEND	IRWRT	002004R	003 READY	002616RG	WTKG	001552R	
B1T03	= 000010	DATA	001212R	IRWTTM	IRWTTM	001000R	002 ROM	002466R	WTKEN	001474RG	
B1T04	= 000020	DESTR	001256RG	LIDFD	LIDFD	001400R	002 RST	002464R	WTRAM	001710R	
B1T05	= 000040	DHOST =	000001	LIO	LIO	000430RG	003 RSTART	002414RG	XCIO =	000001	
B1T06	= 000100	DMAOK	001160R	LIUINT	LIUINT	000076RG	RS0	002420R	XGIO =	000001	
B1T07	= 000200	EMBF	001024R	LLFLG	LLFLG	002002R	RS1	002432R	XINBF	000400R	003
B1T08	= 000400	ENABLE	001214RG	LPINFT=	LPINFT=	***** G	RS2	002452R	XOUTBF	000000R	003
B1T14	= 040000	END\$	001202R	LPOUT	LPOUT	001246RG	RTI\$	001000R	XSID =	000000	
B1T15	= 100000	ENSTR	001246RG	LTIME	LTIME	001006R	SIO	002470RG	ZER08F	000304R	
B0\$	000644R	E0\$	001226R								

* ABS. 000000 000
002642 001
DFM 001010 002
BUFS 002006 003
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 51 PAGES
*DK:FDM=DK:FDM

AD-A078 391

BURROUGHS CORP PAOLI PA FEDERAL AND SPECIAL SYSTEMS GROUP F/6 9/2
SOFTWARE MAINTENANCE MANUAL FOR THE MODULAR SYSTEM CONTROL DEVE--ETC(U)
NOV 79 DCA100-76-C-0083

UNCLASSIFIED

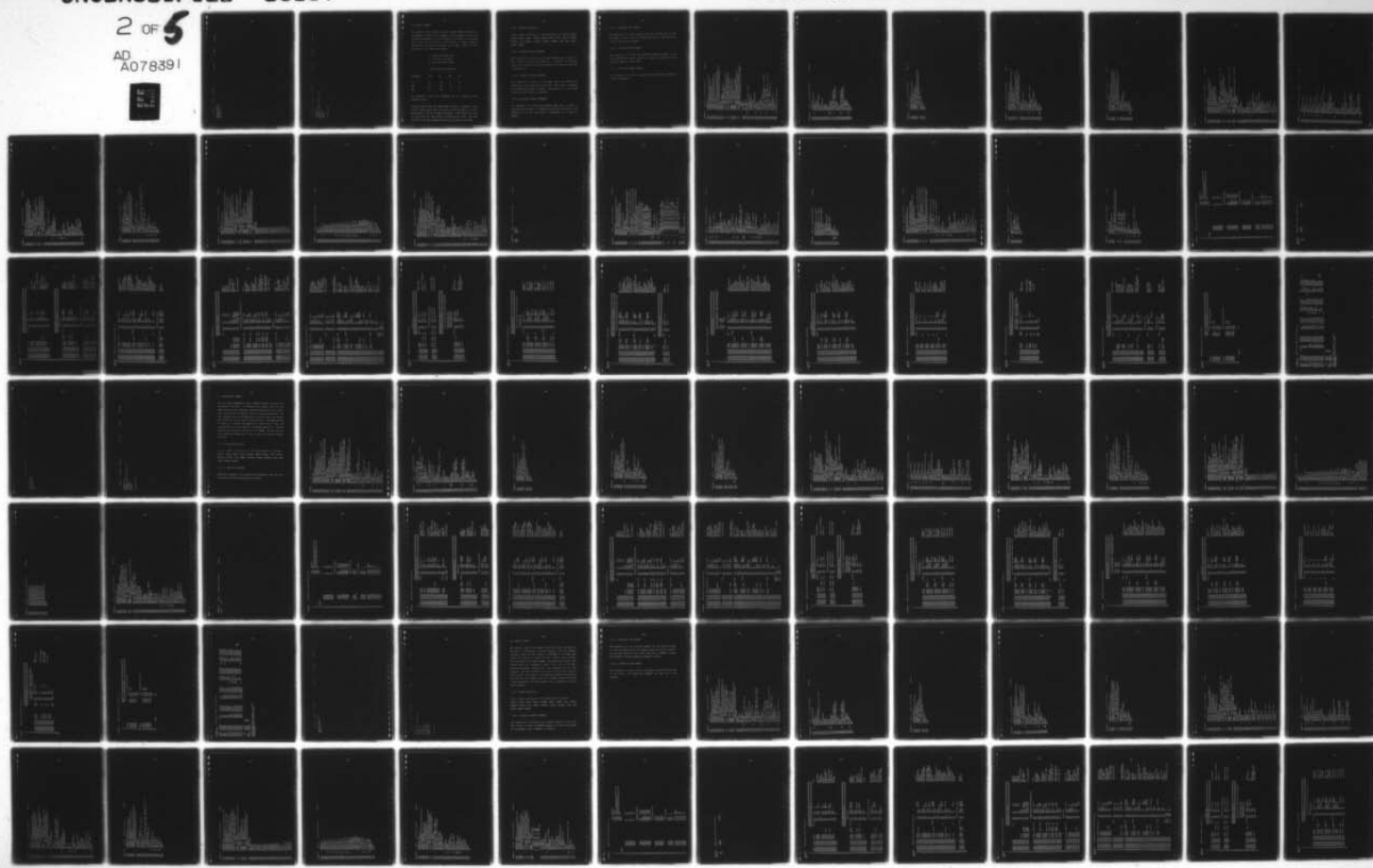
66157

SBIE-AD-E100 313

NL

2 OF 5

AD A078391



decipher

program:

CONF23.COM

01-JUL-79

12:00:00

PAGE 001

FOR NODAL /NOD INF
FOR NODAL 1/NOD INF
FOR 0-000/NOD INF
MAC FID

PAGE 001

12:00:00

01-JUL-79

PRG:00100

PROGRAM:

devul. for

ASSIGN DXO: RN:
 E LINK
 DXI:NODE.22-DX1:MAP DX1:NODAL/C/I/W
 DX1:100/F
 DX1:NODAL/C
 DX1:VSURE//
 \$START

ASSIGN DX1: RN:
 03516N

1.4 NODE 23 (DSQC).

The Digital Service Quality Control (DSQC) module assesses the performance of digital data channels for the purpose of detecting degrading performance of these channels with respect to increasing error rates, and to assist in fault isolation. Channels numbered 501-1000 are arbitrarily monitored by the DSQC. There are three parameters to be checked per channel:

1. Pseudo Error Rate (PE)
2. Bit Error Rate (BIE)
3. Block Error Rate (BLE)

DSQC Parameters Thresholds

<u>Parameter</u>	<u>RH</u>	<u>RL</u>	<u>AH</u>	<u>AL</u>
PE	10	-10	5	-5
BIE	10	-10	5	-5
BLE	10	-10	5	-5

The parameter values are simulated by the Simulated Input Generator (SIG).

Trending occurs when the DWQC module detects a parameter value within a delta value equal to one of the Red threshold or a delta value equal to two of the Amber threshold. Event reports are sent to FIAC using the same format as described for VSQC. The DSQC interprets the same commands generated by the DBMS as the VSQC.

1.4.1 Program Descriptions

1.4.1.1 Refer to Section 1.1 for descriptions of routines NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME RAM, LPOUT, RDPNT, RSTART.

1.4.1.2 Subroutine DSQC (FORTRAN)

This subroutine is called from the nodal program when a packet has been received from the SIG interface. It decodes the simulated input values read from the SIG and performs the function described in Section 1.4.

1.4.1.3 Subroutine TREND (FORTRAN)

This subroutine is called by the DSQC routine to perform the trending on the values sent by the SIG. This value is compared with DELTA values; when a channel consistently is in the DELTA region, an event report is generated.

1.4.1.4 Subroutine DSQCLP (FORTRAN)

The subroutine is called from program NODAL when a packet is received from the loop. It performs one of three functions: turns reporting on or off, and takes a measurement of a specified channel.

1.4.1.5 Subroutine SIO (MACRO)

The subroutine is called from an interrupt received by the SIG interface; it then reads the message and sets a flag containing the byte count of the message.

1.4.1.6 Subroutine SOUT (MACRO)

This subroutine is called from subroutine DSQC and passes to the SIG processor the channel number of a channel for which the user language requests a measurement.

1.4.1.7 Subroutine READY (MACRO)

This subroutine is called to signal the SIG that DSQC is ready for another measurement.

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:47:35      PAGE 001

0001      PROGRAM MODAL
0002      INTEGER*2 XING,XOUTQ,ACKQ,PXING,FXOUTQ,PACKQ
0003      INTEGER*2 PING,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0007      LOGICAL*1 IDATA
0008      LOGICAL*1 PACK,ICODE,ETX,CR,LF,MONTOR,ISLID,DUM
0009      INTEGER*2 LTIME,ACKTIM,NEWTIM,OLDTIM,IWRTIM,TIMLIM,ATIMLM
0010      REAL*4 RH,RL,AH,AL,RHT,RLT,AHT,ALT,DELTA,IRSV,VMEAS
0011      REAL*8 VTR
0012      COMMON /MESS/ MESSEQ
0013      COMMON /DFM/ OUTBF(256),INBF(256),
0014      1 IWRTIM,IWRT,STAT,LTIME
0015      1 COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),FXOUTQ(2),
0016      1 ACKQ(16),PACKQ(2),INQ(16),PING(2)
0017      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0018      1 LIDFD(256),IOFLG,LLFLG,IRSEND
0019      1 COMMON /FRE/ FREE(64),IFR,IFRSZ
0020      1 COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0021      1 COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0022      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0023      1 RESNLM,IALIRT,ISLID,MONTOR,LF,ETX,CR,DUM
0024      1 COMMON /DCOM/ ISWTRCH,VMEAS(3),ICTR,RL(3),RH(3),AL(3),AH(3),
0025      1 ITCM,NFL,SFL,VTR,IVC,MON,IDATA(10),IRSV(3),ICHNSW,
0026      2 ICHAN
0027      CALL MASTER
0028      CALL INIT
0029      CALL ENABLE(1)
0030      5 CALL READY
0031      25 IF(PING(1).LT. PING(2)) GOTO 40
0032      CALL ENABLE(0)
0033      Q2=DEQUE(PING,INQ,1)
0034      CALL ENABLE(1)
0035      DO 30 I=1,10
0036      1 DATA(I)=PACK(I+6,Q2)
0037      CONTINUE
0038      CALL DSQCLP
0039      IFR=IFR+1
0040      FREE(IFR)=Q2
0041      CALL STATB0(15)
0042      IF(IS.EQ.1) GOTO 100
0043      IF(IOFLG.GT. 0 .AND. IRSEND.EQ. 0) CALL DSQC
0044      IF(LLFLG.GT. 0) CALL LINLOS
0045      80 IF(OUTFCT.EQ. 1) GOTO 100
0046      IF(IFULL.EQ. 1) .OR. (ISENT.EQ. 1)) GOTO 100
0047      IF(IRSEND.EQ. 1) GOTO 85
0048      IF(PXING(1).LT. PXING(2)) GOTO 100
0049      CALL ENABLE(0)
0050      Q1=DEQUE(PXING,XING,1)
0051      CALL ENABLE(1)
0052      OUTQ=Q1
0053      85 CALL DESTR(PACK(254,OUTQ),Q2)

```

PAGE 002

FORTRAN IV U02.1-1 Mon 04-Jun-79 17:47:35

```

0057 DO 90 I=1,Q2
0058   OUTBF(I)=PACK(I,OUTQ)
0059   CONTINUE
0060   OUTBF(Q2+1)=0
0061   IPT=OUTBF(5)
0062   OUTBF(Q2+2)=LIDFD(IPT)
0063   IFULL=1
0064   ISENT=1
0065   CALL LPOUT(Q2+2)
0066   IRSEND=1
0067   ACKTIM=0
0068   IWRTIM=0
0069   IWRT=1
0070   INFCT=0
0071   100 OLDTIM=NEWTIM
0072   NEWTIM=LTIME
0073   IF(IWRT.EQ. 0) GOTO 120
0075   IWRTIM=IWRTIM+(NEWTIM-OLDTIM)
0076   IF(IWRTIM.LT. TIMLM) GOTO 120
0078   CALL WOKEN
0079   IWRTIM=0
0080   IWRT=0
0081   120 IF(ISENT.EQ. 0) GOTO 130
0083   ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0084   IF(ACKTIM.LT. ATIMLM) GOTO 130
0086   CALL ENABLE(0)
0087   CALL ACKNAK(0)
0088   CALL ENABLE(1)
0089   130 CONTINUE
0090   GOTO 5
0091   END

```

PAGE 001

FORTTRAN IV V02.1-1 Mon 04-Jun-79 17:47:55

```

0001 FUNCTION IGETSP(N)
0002 LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003 INTEGER*2 FREE,FLWCNT
0004 INTEGER*2 OUTFCT,OUTQ,RESNLM
0005 COMMON /FRE/ FREE(64),IFR,IFRSZ
0006 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
      1 RESNLM,IALIRT,ISLID,MONITOR,LF,ETX,CR,DUM
0007 IF (IFR .LT. 1) CALL INIT
0008 IGETSP=FREE(IFR)
0009 IFR=IFR-1
0010 RETURN
0011 END
0012

```

PAGE 001

FORTTRAN IV V02.1-1 Mon 04-Jun-79 17:48:07

```

0001 SUBROUTINE ENQUE(A,B,N)
0002 LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003 INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWCNT,OUTQ,A(2),B(16)
0005 INTEGER*2 RESNLM,OUTFCT
0006 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007      1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0008      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0010      10 IQHEAD=A(1)
0011      IQTAIL=A(2)
0012      IF(IQTAIL.EQ.1) GOTO 20
0013      IQTAIL=IQTAIL-1
0014      B(IQTAIL)=N
0015      A(2)=IQTAIL
0016      GOTO 999
0017      20 IF(IQHEAD.GE.(IQLNTH)) GOTO 40
0018      NN=IQHEAD-IQTAIL
0019      DO 30 I=1,NN+1
0020      30 B(IQLNTH+1-I)=B(IQHEAD+1-I)
0021      A(1)=IQLNTH
0022      A(2)=IQLNTH-NN
0023      GOTO 10
0024      40 CALL INIT
0025      999 RETURN
0026      END

```



```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:48:20      PAGE 001

0001      FUNCTION DEQUE(A,B,IO)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOR
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,FIND
0004      INTEGER*2 FLWCNT,A(2),B(16),DEQUE,OUTFCT,OUTQ,RESNM
0005      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      1      ACKQ(16),PACKQ(2),INQ(16),FINQ(2)
0007      1      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1      RESNM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0009      IOHEAD=A(1)
0010      DEQUE=B(IOHEAD)
0011      IF(ID.NE.1) GOTO 999
0012      IF(IOHEAD.NE.0) GOTO 10
0013      A(1)=IQLNTH
0014      A(2)=IQLNTH+1
0015      GOTO 999
0016      10 A(1)=IOHEAD-1
0017      999 RETURN
0018      END

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:48:41      PAGE 001

0001      SUBROUTINE ACKNAK(N)
0002      INTEGER*2 FLWcnt,OUTFCT,STAT,XINQ,PXINQ,PXOUTQ
0003      INTEGER*2 ACKQ,PACKQ,PINQ,FREE,OUTQ,XOUTQ
0004      INTEGER*2 I1,T2,T3,T4,T5,RESNLM
0005      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0006      LOGICAL*1 PACK,OUTBF,INBF,LCOMT(40)
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0008      INTEGER*2 LTIME,IWRITM
0009      REAL*8 COMT(5)
0010      EQUIVALENCE(COMT,LCOMT)
0011      DATA COMT(1)/'MESSAGE '//,COMT(2)/'NOT SENT',//,COMT(3)/' FROM '//,
0012      1 COMT(4)/'NODE '//,COMT(5)/' TO NODE'//
0013      1 COMMON /GLOB/ ISENT,FLWcnt,IQLNTH,OUTFCT,IFULL,OUTQ,
0014      1 RESNLM,IALIRT,ISLID,MONITOR,LF,ETX,CR,DUM
0015      1 COMMON /DFM/ OUTBF(256),INBF(256),
0016      1 IWRITM,IWRT,STAT,LTIME
0017      1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0018      1 PACKQ(2),INQ(16),PINQ(2)
0019      1 COMMON /FRE/ FREE(64),IFR,IFRSZ
0020      1 COMMON /MESS/ MESSED
0021      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0022      1 IOFLG,LLFLG,IRSEND
0023      IF (N.NE. 1) GOTO 10
0024      IFR=IFR+1
0025      DO 5 I=1,256
0026      PACK(I,OUTQ)=0
0027      IFULL=0
0028      ISENT=0
0029      IRSEND=0
0030      FREE(IFR)=OUTQ
0031      GOTO 999
0032      T2=PACK(256,OUTQ)
0033      IF(T2.LE. RESNLM) GOTO 120
0034      T3=PACK(3,OUTQ)
0035      T4=MOD(T3,64)
0036      IF(T4.LT. 32) GOTO 110
0037      T5=IGETSP(N)
0038      IF(MESSED.EQ. 126) MESSED=0
0039      MESSED=MESSED+1
0040      PACK(1,T5)=0
0041      PACK(2,T5)=MESSED
0042      PACK(3,T5)=0
0043      PACK(4,T5)=0
0044      PACK(5,T5)=25
0045      PACK(6,T5)=ISLID
0046      DO 20 I=7,60
0047      PACK(I,T5)=*040
0048      20 CONTINUE
0049      DO 30 I=7,9
0050      PACK(I,T5)=LF
0051      30 CONTINUE

```

PAGE 002

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:48:41

```

0054      DO 40 I=1,8
0055         PACK(I+9,T5)=LCOMT(I)
0056      CONTINUE
0057      DO 50 I=9,16
0058         PACK(I+9,T5)=LCOMT(I)
0059      CONTINUE
0060      DO 60 I=17,22
0061         PACK(I+9,T5)=LCOMT(I)
0062      CONTINUE
0063      DO 70 I=25,29
0064         PACK(I+7,T5)=LCOMT(I)
0065      CONTINUE
0066      ENCODE(3,80,PACK(38,T5)) ISLID
0067      FORMAT(I3)
0068      DO 90 I=33,40
0069         PACK(I+11,T5)=LCOMT(I)
0070      CONTINUE
0071      ENCODE(3,80,PACK(52,T5))PACK(5,OUTQ)
0072      PACK(55,T5)=LF
0073      PACK(56,T5)=LF
0074      PACK(57,T5)=CR
0075      PACK(58,T5)=ETX
0076      CALL ENSTR(PACK(254,T5),58)
0077      CALL ENQUE(PXING,XING,T5)
0078      DO 102 I=1,256
0079         PACK(I,OUTQ)=0
0080      CONTINUE
0081      ISENT=0
0082      IRSEND=0
0083      IFULL=0
0084      IWRT=0
0085      IFR=IFR+1
0086      FREE(IFR)=OUTQ
0087      GOTO 999
0088      110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089      PACK(256,OUTQ)=0
0090      ISENT=0
0091      IFULL=0
0092      IF(IRSEND.EQ.1) GOTO 999
0094      CALL ENQUE(PXING,XING,OUTQ)
0095      GOTO 999
0096      120 PACK(256,OUTQ)=T2
0097      IFULL=0
0098      ISENT=0
0099      IF(IRSEND.EQ.1) GOTO 999
0101      CALL ENQUE(PXING,XING,OUTQ)
0102      RETURN
0103      END

```

PAGE 001

FORTRAN IV V02.1-1 Mon 04-Jun-79 17:49:03

```

0001 SUBROUTINE INPTQ(L)
0002 INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,FXOUTQ,ACKQ
0003 INTEGER*2 PACKQ,PINQ,FLWCNT,OUTFCT,I1,RESNLM,OUTQ
0004 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0005 LOGICAL*1 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
0006 INTEGER*2 LTIME,IWRITM
0007 COMMON /DFM/ OUTBF(256),INBF(256),
0008      IWRITM,IWRT,STAT,LTIME
0009 COMMON /RFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010      LIDFD(256),IOFLG,LLFLG,IRSEND
0011 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0013 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0014      RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0015 IF(L.LT. 1) GOTO 30
0016 LI=L-2
0017 II=INBF(6)
0018 IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
0019 T1=IGETSP(N)
0020 DO 10 I=1,LI
0021 PACK(I,T1)=INBF(I)
0022 CALL ENSTR(PACK(254,T1),LI)
0023 CALL ENQUE(PINQ,INQ,I1)
0024 N=2
0025 GOTO 40
0026 N=1
0027 GOTO 40
0028 N=0
0029 DO 50 I=1,2
0030      OUTBF(I)=INBF(I)
0031 CONTINUE
0032 NN=1
0033 IF(N.EQ. 0) NN=128
0034 OUTBF(3)=NN
0035 OUTBF(4)=0
0036 OUTBF(5)=INBF(6)
0037 OUTBF(6)=ISLID
0038 OUTBF(7)=EXT
0039 OUTBF(8)=0
0040 OUTBF(9)=LIDFD(INBF(6))
0041 CALL LPOUT(9)
0042 IF(N.NE. 2) GOTO 999
0043 ACKSEQ(II)=INBF(2)
0044 RETURN
0045 999
0046 END

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:49:19      PAGE 001

0001      SUBROUTINE LPIMPT(LI)
0002      INTEGER*2 STAT,FLWCNT,OUTFCT,OUTO,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTRF,INBF,ETX,CR,LF,MONTR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRTTH
0006      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTO,
0007      1      RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0008      CC1=INBF(3)
0009      CC2=INBF(4)
0010      IF(CC1.LT. 0) GOTO 25
0012      IF(CC1.EQ. 0 .AND. CC2.EQ. 0) GOTO 40
0014      CC3=MOD(CC1,2)
0015      CC4=MOD(CC1,256)
0016      IF ((CC3.GE. 1 .OR. CC4.GE. 128) .AND. ISENT.EQ. 1) GOTO 20
0018      IF ((INBF(1).EQ. 85) .AND. INBF(2).EQ. 170) GOTO 99
0020      CC5=MOD(INBF(3),64)
0021      IF(CC5.GE. 32) CALL INPTQ(LI)
0023      GOTO 99
0024      20      CC7=MOD(CC1,2)
0025      IF(CC7.GE. 1) GOTO 30
0027      25      CALL ACKNAK(0)
0028      GOTO 99
0029      30      CALL ACKNAK(1)
0030      GOTO 99
0031      40      CALL INPTQ(LI)
0032      99      CONTINUE
0033      RETURN
0034      END

```



```

FORTRAN IV      V02.1-1      Mon. 04-Jun-79 17:49:34      PAGE 001

0001      SUBROUTINE INIT
0002      REAL*4 RH,RL,AM,AL,IRSV,VMEAS
0003      REAL*8 VTR
0004      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ,FREE
0005      INTEGER*2 STAT,FLWCNT,OUTFCT
0006      INTEGER*2 SETPRM,RSTPRM,SETBKP,OUTQ,RESNLM,RSTBKP
0007      LOGICAL*1 DUM,IDATA
0008      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0009      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID
0010      INTEGER*2 LTIME,OLDTIM,TIMLIM,ACKTIM,ATIMLM,IWRTTM
0011      COMMON /DFM/ OUTBF(256),INBF(256),
1          IWRTTM,IWRT,STAT,LTIME
0012      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
1          ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0013      COMMON /MESS/ MESSEQ
0014      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1          IOFLG,LLFLG,IRSEND
0015      COMMON /FRE/ FREE(64),IFR,IFRSZ
0016      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0017      COMMON
0018      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0019      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0020      COMMON /DCOM/ ISWITCH,VMEAS(3),ICTR,RL(3),RH(3),AL(3),AH(3),
1          ITERM,NEL,SFL,VTR,IVC,MON,IDATA(10),IRSV(3),ICHNSW,
2          ICHAN
0021      CALL RAM(0,6,4)
0022      CALL RAM(0,255,0)
0023      MESSEQ=0
0024      IOFLG=0
0025      IRSEND=0
0026      LLFLG=0
0027      IFRSZ=64
0028      INFLCT=0
0029      ISLID=23
0030      ICHNSW=0
0031      CR='015'
0032      LF='012'
0033      ETX='003'
0034      RESNLM=2
0035      IQLNTH=16
0036      IALTRT=0
0037      FLWCNT=0
0038      IWRTTM=0
0039      TIMLIM=50
0040      SETPRM=4
0041      RSTPRM=32
0042      SETBKP=8
0043      RSTBKP=64
0044      ATIMLM=200
0045      MONTOR=27
0046      ICHNSW=0
0047      IWRT=0
0048      ISWITCH=0

```

PAGE 002

Mon 04-Jun-79 17:49:34

V02.1-1

FORTRAN IV

```

0049 OUTFACT=0
0050 IFULL=0
0051 ISENT=0
0052 DO 40 I=1,20
0053   LIDFD(I)=4
0054   LIDFD(21)=1
0055   LIDFD(22)=3
0056   LIDFD(23)=6
0057   LIDFD(24)=5
0058   LIDFD(25)=7
0059   LIDFD(26)=8
0060   LIDFD(27)=9
0061   LIDFD(28)=2
0062   DO 50 I=29,39
0063     LIDFD(I)=0
0064     DO 60 I=40,44
0065       LIDFD(I)=1
0066       DO 70 I=45,59
0067         LIDFD(I)=0
0068         DO 80 I=60,64
0069           LIDFD(I)=2
0070           DO 90 I=65,79
0071             LIDFD(I)=0
0072             DO 100 I=80,84
0073               LIDFD(I)=4
0074               DO 110 I=85,99
0075                 LIDFD(I)=0
0076                 DO 120 I=100,104
0077                   LIDFD(I)=5
0078                   DO 130 I=105,256
0079                     LIDFD(I)=0
0080                     PING(1)=IQLNTH
0081                     PING(2)=IQLNTH+1
0082                     PACKQ(1)=IQLNTH
0083                     PACKQ(2)=IQLNTH+1
0084                     PXOUTQ(1)=IQLNTH
0085                     PXOUTQ(2)=IQLNTH+1
0086                     PXLNTH(1)=IQLNTH
0087                     PXLNTH(2)=IQLNTH+1
0088                     DO 10 I=1,IFRSZ
0089                       PACK(255,I)=0
0090                       DO 20 I=1,IFRSZ
0091                         FREE(I)=I
0092                         DO 30 I=1,256
0093                           ACKSEQ(I)=256
0094                           IFR=IFRSZ
0095                           RETURN
0096                           END
0097

```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 17:49:55      PAGE 001

0001      SUBROUTINE LINLOS
0002      REAL*8 RM1(5),LIN08,LIN18
0003      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM,T1
0004      INTEGER*2 XINQ,PXINQ,XOUTQ,FXOUTQ,ACKQ,PACKQ,PINQ
0005      LOGICAL*1 ETX,CR,LF,MONTR,ISLID,DUM
0006      LOGICAL*1 PACK,M1(40),LINO(8),LINI(8)
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0008      COMMON/QUE/XINQ(16),PXINQ(2),XOUTQ(16),FXOUTQ(2),ACKQ(16),
0009      *      PACKQ(2),INO(16),PINQ(2)
0010      COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0011      *      IOFLG,LLFLG,IRSEND
0012      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0013      *      RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0014      COMMON /MESS/ MESSEQ
0015      DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0016      DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0017      DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0018      DATA RM1(5)/'DE 23 '//
0019      EQUIVALENCE(LINO,LIN08)
0020      EQUIVALENCE(LINI,LIN18)
0021      EQUIVALENCE(M1,RM1)
0022      IS=LLFLG
0023      CALL ENABLE(0)
0024      T1=IGETSP(N)
0025      CALL ENABLE(1)
0026      PACK(1,T1)=0
0027      IF(MESSEQ.EQ.126) MESSEQ=0
0028      MESSEQ=MESSEQ+1
0029      PACK(2,T1)=MESSEQ
0030      PACK(3,T1)=0
0031      PACK(4,T1)=0
0032      PACK(5,T1)=25
0033      PACK(6,T1)=23
0034      DO 20 I=7,9
0035      *      PACK(I,T1)=LF
0036      20 CONTINUE
0037      DO 30 I=1,22
0038      *      PACK(I+9,T1)=M1(I)
0039      IF(IS.EQ.1) GOTO 50
0040      DO 40 I=1,8
0041      *      PACK(I+31,T1)=LINO(I)
0042      GOTO 70
0043      50 DO 60 I=1,8
0044      *      PACK(I+31,T1)=LINI(I)
0045      DO 80 I=23,40
0046      *      PACK(I+17,T1)=M1(I)
0047      PACK(58,T1)=CR
0048      PACK(59,T1)=LF
0049      PACK(60,T1)=ETX
0050      CALL ENSTR(PACK(254,T1),60)
0051      CALL ENABLE(0)
0052      CALL ENQUE(PXINQ,XINQ,T1)
0053      CALL ENABLE(1)

```

PAGE 002

Mon 04-Jun-79 17:49:55

V02.1-1

FORTRAN IV

0054 LLFLG=0
0055 RETURN
0056 END

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 18:07:15      PAGE 001

0001      SUBROUTINE DSQC
0002      INTEGER*2 XING,XOUTQ,ACKQ,PXING,PXOUTQ,PACKQ,PING
0003      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,IDATA
0005      LOGICAL*1 PACK,ISLID,MONTOR,LF,ETX,CR,DUM
0006      REAL*8 VTR
0007      REAL*4 RH,RL,AH,AL,VMEAS,IRSV
0008      COMMON PACK(256,64)
0009      COMMON /MESS/ MESSEQ
0010      COMMON /DCOM/ ISWITCH,VMEAS(3),ICTR,RL(3),RH(3),AL(3),AH(3),
1          ITERN,NFL,SFL,VTR,IVC,MON,IDATA(10),IRSV(3),ICHNSW,
2          ICHAN
0011      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1          IOFLG,LLFLG,IRSEND
0012      COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
1          PACKQ(2),INQ(16),PING(16)
0013      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,RESNLM,
1          IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0014      DATA RH(1)/40.5/,RH(2)/19.9/,RH(3)/45.5/
0015      DATA RL(1)/-40.5/,RL(2)/-16.9/,RL(3)/-45.5/
0016      DATA AH(1)/35.5/,AH(2)/16.5/,AH(3)/40.5/
0017      DATA AL(1)/-35.5/,AL(2)/-16.5/,AL(3)/-40.5/
0018      DECODE(2,20,XINBF(1),ERR=410) ICNT
0019      DECODE(4,30,XINBF(3),ERR=410) VMEAS(1)
0020      DECODE(4,30,XINBF(7),ERR=410) VMEAS(2)
0021      DECODE(4,30,XINBF(11),ERR=410) VMEAS(3)
0022      DECODE(2,20,XINBF(15),ERR=410) IVC
0023      DECODE(2,20,XINBF(25),ERR=410) MON
0024      FORMAT(A2)
0025      30      FORMAT(A4)
0026      IF(ICHNSW.EQ. 1) GOTO 400
0028      DO 50 I=1,3
0029      IRSV(I)=(AH(I)+AL(I)) / 2
0030      50      CONTINUE
0031      DO 350 ICTR=1,3
0032      IF(VMEAS(ICTR).GE. (AL(ICTR)+2) .AND.
1          VMEAS(ICTR).LE. (AH(ICTR)-2)) GOTO 350
0034      IF ((VMEAS(ICTR).GT. (AH(ICTR)-2) .AND.
1          VMEAS(ICTR).LT. AH(ICTR)) .OR.
2          (VMEAS(ICTR).GT. AL(ICTR) .AND.
3          VMEAS(ICTR).LT. (AL(ICTR)+2))) GOTO 100
0036      IF ((VMEAS(ICTR).GE. AH(ICTR) .AND.
1          VMEAS(ICTR).LE. (RH(ICTR)-1)) .OR.
2          (VMEAS(ICTR).GE. (RL(ICTR)+1) .AND.
3          VMEAS(ICTR).LE. AL(ICTR))) GOTO 200
0038      IF ((VMEAS(ICTR).GT. (RH(ICTR)-1) .AND.
1          VMEAS(ICTR).LT. RH(ICTR)) .OR.
2          (VMEAS(ICTR).GT. RL(ICTR) .AND.
3          VMEAS(ICTR).LT. (RL(ICTR)+1))) GOTO 300
0040      IF(ISWITCH.EQ. 1) GOTO 40
0042      GOTO 410
0043      40      ISTAT=1
0044      GOTO 210
0045      100      IF(ISWITCH.EQ. 1) GOTO 110

```


FORTRAN IV V02.1-1-1 Mon 04-Jun-79 18:07:15

```

0047      GOTO 410
0048 110  CALL TREND(2,VMEAS(ICTR))
0049      GOTO 410
0050 200  ISTAT=2
0051      IF(ISTWCH .EQ. 1) GOTO 210
0052      GOTO 410
0053 210  CALL ENABLE(0)
0054      K1=IGETSP(N)
0055      CALL ENABLE(1)
0056      IF(MESSEQ .EQ. 126) MESSEQ=0
0057      MESSEQ=MESSEQ+1
0058      PACK(1,K1)=0
0059      PACK(2,K1)=MESSEQ
0060      PACK(3,K1)=0
0061      PACK(4,K1)=0
0062      PACK(5,K1)=MONTOR
0063      PACK(6,K1)=ISLID
0064      DO 250 I=7,14
0065          PACK(I,K1)=XINBF(I+10)
0066      CONTINUE
0067 250  DO 260 I=15,16
0068          PACK(15,K1)=XINBF(I)
0069      CONTINUE
0070 260  PACK(17,K1)=ISTAT
0071      ENCODE(4,270,PACK(18,K1)) VMEAS(ICTR)
0072      ENCODE(2,280,PACK(22,K1)) MON
0073      FORMAT(A4)
0074      FORMAT(A2)
0075      PACK(24,K1)=ITERM
0076      CALL ENSTR(PACK(254,K1),24)
0077      CALL ENABLE(0)
0078      CALL ENQUE(PXIND,XIND,K1)
0079      CALL ENABLE(1)
0080      GOTO 410
0081 300  IF(ISTWCH .EQ. 1) GOTO 310
0082      GOTO 410
0083 310  CALL TREND(2,VMEAS(ICTR))
0084      GOTO 410
0085 350  CONTINUE
0086      GOTO 410
0087      CALL ENABLE(0)
0088      K1=IGETSP(N)
0089      CALL ENABLE(1)
0090      IF(MESSEQ .EQ. 126) MESSEQ=0
0091      MESSEQ=MESSEQ+1
0092      PACK(1,K1)=0
0093      PACK(2,K1)=MESSEQ
0094      PACK(3,K1)=0
0095      PACK(4,K1)=0
0096      PACK(5,K1)=25
0097      PACK(6,K1)=ISLID
0098      PACK(7,K1)=CR
0099      PACK(8,K1)=LF
0100      ENCODE(4,360,PACK(9,K1)) ICHAN
0101
0102
0103
0104

```

PAGE 003

FORTRAN IV V02.1-1 Mon 04-Jun-79 18:07:15

```

0105  PACK(13,K1)=*40
0106  ENCODE(12,370,PACK(14,K1)) VMEAS(1)
0107  PACK(26,K1)=*40
0108  ENCODE(12,370,PACK(27,K1)) VMEAS(2)
0109  PACK(39,K1)=*40
0110  ENCODE(12,370,PACK(40,K1)) VMEAS(3)
0111  PACK(52,K1)=CR
0112  PACK(53,K1)=LF
0113  PACK(54,K1)=ETX
0114  CALL ENSTR(PACK(254,K1),54)
0115  CALL ENABLE(0)
0116  CALL ENQUE(PXING,XING,K1)
0117  CALL ENABLE(1)
0118  ICHNSW=0
0119  FORMAT(I4)
0120  360  FORMAT(F12.6)
0121  410  IOFLG=0
0122  RETURN
0123  END

```

```

FORTAN IV      V02.1-1      Mon 04-Jun-79 18:07:41      PAGE 001

0001      SUBROUTINE TREND(ICND,IVAL)
0002      REAL*4 RH,RL,AH,AL,VMEAS,IRSV,IVAL
0003      REAL*8 VTR
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,IDATA
0005      LOGICAL*1 PACK,ISLID,MONITOR,LF,ETX,CR,DUM
0006      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ
0007      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM
0008      COMMON PACK(256,64)
0009      COMMON /MESS/ MESSEQ
0010      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0011      1      PACKQ(2),INQ(16),PINQ(2)
0012      1      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,RESNLM,
0013      1      IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0014      1      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0015      1      IOFLG,LLFLG,IRSEND
0016      1      COMMON /DCOM/ ISWITCH,VMEAS(3),ICTR,RL(3),RH(3),AL(3),AH(3),
0017      1      ITERM,NFL,SFL,VTR,IVC,MON,IDATA(10),IRSV(3),ICHNSW,
0018      2      ICHAN
0019      IU=(RH(ICTR)-RL(ICTR))/30
0020      IF((IRSV(ICTR)-IVAL).GT. (4 * IU)) GOTO 20
0021      IF(NFL.GE. 3) GOTO 40
0022      IF((IRSV(ICTR)-IVAL).GT. (2 * IU)) GOTO 30
0023      NFL=NFL+1
0024      10      IRSV(ICTR)=(.33 * IVAL + .67 * IRSV(ICTR))
0025      GOTO 60
0026      20      IF(SFL.GE. 2) GOTO 30
0027      SFL=SFL+1
0028      GOTO 50
0029      30      IRSV(ICTR)=IVAL
0030      NFL=0
0031      GOTO 50
0032      40      SFL=0
0033      GOTO 10
0034      50      IF(ICND .EQ. 2) RETURN
0035      60      CALL ENABLE(0)
0036      K1=IGETSP(N)
0037      CALL ENABLE(1)
0038      IF(MESSEQ .EQ. 126) MESSEQ=0
0039      MESSEQ=MESSEQ+1
0040      PACK(1,K1)=0
0041      PACK(2,K1)=MESSEQ
0042      PACK(3,K1)=0
0043      PACK(4,K1)=0
0044      PACK(5,K1)=MONITOR
0045      PACK(6,K1)=ISLID
0046      DO 70 I=7,14
0047      70      PACK(I,K1)=XINBF(I+10)
0048      CONTINUE
0049      DO 75 I=15,16
0050      75      PACK(I,K1)=XINBF(I)
0051      ENCODE(4,80,PACK(18,K1)) IVAL
0052      FORMAT(A4)
0053      PACK(17,K1)=ICND
0054

```

PAGE 002

FORTRAN IV V02.1-1 Mon 04-Jun-79 18:07:41

```
0056 90 ENCODE(2,90,PACK(22,K1)) MON
0057 90 FORMAT(A2)
0058 90 PACK(24,K1)=ITERM
0059 90 CALL ENSTR(PACK(254,K1),24)
0060 90 CALL ENABLE(0)
0061 90 CALL ENQUE(PXING,XING,K1)
0062 90 CALL ENABLE(1)
0063 90 RETURN
0064 90 END
```

```

FORTRAN IV      V02.1-1      Mon 04-Jun-79 18:07:59      PAGE 001

0001      SUBROUTINE DSQCLF
0002      LOGICAL*1 IDATA
0003      REAL*4 RH,RL,AL,UMEAS,IRSV
0004      REAL*8 VTR
0005      COMMON /DCOM/ ISWITCH,UMEAS(3),ICTR,RL(3),RH(3),AL(3),AH(3),
1      ITERM,NEL,SFL,VTR,IVC,MON,IDATA(10),IRSV(3),ICHNSW,
2      ICHAN
0006      IF(IDATA(1).EQ.'117'.AND.
1      IDATA(2).EQ.'116') GOTO 50
0008      IF(IDATA(1).EQ.'117'.AND.
1      IDATA(2).EQ.'106'.AND.
2      IDATA(3).EQ.'106') GOTO 60
0010      IF(IDATA(1).EQ.'115') GOTO 70
0012      GOTO 80
0013      ISWITCH=1
0014      ITERM=IDATA(4)
0015      RETURN
0016      ISWITCH=0
0017      RETURN
0018      70 DECODE(4,75,IDATA(2),ERR=80) ICHAN
0019      75 FORMAT(I4)
0020      CALL SOUT(ICHAN)
0021      ICHNSW=1
0022      RETURN
0023      END

```



```

1  .TITLE FDM.MACRO
2  .SBTTL NODE 23
3  .IDENT /V3.0/
4  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WOKEN,RAM,STATUS,STATBO
6  .GLOBL LFOUT,KSTART,RDPNT,DESTK,ENSTR,LPINPT
7  .NLIST CND
8  .PSECT
9
10 000000
11
12 000000
13 000001
14 000002
15 000003
16 000004
17 000005
18 000006
19 000007
20
21
22 172410
23 172412
24 172414
25 172416
26 172416
27 172416
28 177560
29 177562
30 177564
31 177566
32
33
34
35
36
37
38
39
40
41
42 000001
43 000000
44 000001
45 000000
46 000001
47 000000
48 000001
49
50
51
52 000002
53 000004
54 000006
55 000010
56
57 100000
58 040000
59 000400
60 000200
61 000100
62 000040
63 000020
64 000010
65 000004

R0 =X0
R1 =X1
R2 =X2
R3 =X3
R4 =X4
R5 =X5
SP =X6
PC =X7

;INTERFACE ADDRESS
BAK= 172410
WCR= 172412
CSR= 172414
IDBUF= 172416
OPREG= 172416
HRCR= 177560
HREUF= 177562
HXCSR= 177564
HXBUF= 177566

;INTERFACE OPTIONS
XCIO=1
XSIO=0
XGIO=1
NDXX=0
ND24=1
NHOST=0
DHOST=1

;LOCAL VARIABLES
ARG1= 2
ARG2= 4
ARG3= 6
ARG4= 10

BIT15= 100000
BIT14= 40000
BIT08= 400
BIT07= 200
BIT06= 100
BIT05= 40
BIT04= 20
BIT03= 10
BIT02= 4

```

FIM.MACRO
NODE 23

MACRO V03.01 4-JUN-79 17:50:23 PAGE 1-1

66
67
68

000002
000001

BIT01= 2
BIT00= 1

FDM.MACRO
NODE 23
MACRO V03.01 4-JUN-79 17:50:23 PAGE 3-1

```

58 000722 132767 000002 000254 R1$: R1B: #BIT01, CAUSE
59 000730 001423 REQ RTI$
60 000732 012701 000400' MOV #INBF, R1
61 000736 012702 010610 MOV #4488., R2
62 000742 012703 010640 MOV #4512., R3
63 000746 012704 000002 MOV #BIT01, R4
64 000752 004767 000046 JSR PC, EMBF
65 000756 012705 001206' MOV #AREA, R5
66 000762 012767 000001 MOV #1, AREA
67 000770 010267 000216 MOV R2, DATA
68 000774 004767 000000G JSR PC, LPINPT
69
70 001000 012605 RTI$: MOV (SP)+, R5
71 001002 012604 MOV (SP)+, R4
72 001004 012603 MOV (SP)+, R3
73 001006 012602 MOV (SP)+, R2
74 001010 012601 MOV (SP)+, R1
75 001012 012600 MOV (SP)+, R0
76 001014 052737 040000 #BIT14, @#CSR
77 001022 000002 RTI
78
79 001024 012737 010600 EMBF: MOV #4480., @#OPREG
80 001032 012737 002400 MOV #1280., @#OPREG
81 001040 013700 172416 MOV @#IDBUF, R0
82 001044 130400 BITB R4,R0
83 001046 001002 BNE CRCOK
84 001050 012704 177777 MOV #1, R4
85 001054 010237 172416 MOV R2, @#OPREG
86 001060 012737 001400 MOV #768., @#OPREG
87 001066 105737 172414 TSTB @#CSR
88 001072 100375 BPL -4
89 001074 013702 172416 MOV @#IDBUF, R2
90 001100 042702 177400 BIC #177400, R2
91 001104 010200 MOV R2, R0
92 001106 005400 NEG R0
93 001110 010037 172412 MOV R0, @#WCR
94 001114 010137 172410 MOV R1, @#BAR
95 001120 010337 172416 MOV R3, @#OPREG
96 001124 012737 001400 MOV #768., @#OPREG
97 001132 105737 172414 TSTB @#CSR
98 001136 100375 BPL -4
99 001140 012737 021000 MOV #8704., @#OPREG
100 001146 000240 NOP
101 001150 105737 172414 TSTB @#CSR
102 001154 100401 BMI DMAOK
103 001156 000240 NOP
104 001160 012737 004400 MOV #2304., @#OPREG
105 001166 105737 172414 TSTB @#CSR
106 001172 100375 BPL -4
107 001174 005704 TST R4
108 001176 100001 BPL END$
109 001200 010402 MOV R4, R2
110 001202 000207 RTS FC
111 001204 000000 END$: R15 0
112 001206 000000 CAUSE: .WORD 0
113 001210 000000 AREA: .WORD 0
114 001212 000000 DATA: .WORD 0

```

;INBUF1 FULL
 ;NO WERE DONE
 ;BUFFER ADDRESS
 ;RDBUFADR COMM
 ;SEL INBUF1
 ;CRC BIT
 ;GO EMPTY BUFFER
 ;DATA LINK AREA
 ;ONE VARIABLE
 ;CRC OR BYTE COUNT
 ;CALL FORTRAN QUE'ER

;RESTORE REGISTERS

;ENABLE INTERRUPTS
 ;RETURN FROM INTERRUPT

;READ STATUS 1
 ;RS
 ;FETCH STATUS
 ;GOOD CRC

;NO FLAG
 ;RDBUFADR
 ;RD
 ;GOOD RD

;REPLACE WITH POINTER
 ;CLEAR MST BITS
 ;SAVE IT
 ;2'S COMP

;BYTE COUNT NOW
 ;ADDRESS IN MEMORY
 ;SEL BUFFER
 ;FALSE RD

;DONE ON
 ;NO LOOP
 ;FIRE DMA
 ;DELAY
 ;GOOD DMA

;ERROR IF HERE
 ;FALSE WD
 ;GOOD WD

;WAS CRC OK
 ;YES
 ;NO FLAG IT
 ;RETURN
 ;STATUS BYTE 0 HOLDER

```

1 1 .ENABLE LSB
2 2 *****
3 3 *****
4 4 *****
5 5 *****
6 6 001214 017501 000002 000002 000002 000002 000002 000002 000002 000002
7 7 001220 022701 000001 000001 000001 000001 000001 000001 000001 000001
8 8 001224 001404 000000 000000 000000 000000 000000 000000 000000 000000
9 9 001226 042737 040000 172414 040000 172414 040000 172414 040000 172414
10 10 001234 000207 000000 000000 000000 000000 000000 000000 000000 000000
11 11 001236 052737 040000 172414 040000 172414 040000 172414 040000 172414
12 12 001244 000207 000000 000000 000000 000000 000000 000000 000000 000000
13 13
14 14
15 15
16 16 001246 017575 000004 000002 000004 000002 000004 000002 000004 000002
17 17 001254 000207 000000 000000 000000 000000 000000 000000 000000 000000
18 18 001256 017575 000002 000004 000002 000004 000002 000004 000002 000004
19 19 001264 000207 000000 000000 000000 000000 000000 000000 000000 000000
20 20
21 21
22 22
23 23 *****
24 24 *****
25 25 *****
26 26 *****
27 27 *****
28 28 *****
29 29 *****
30 30 *****
31 31 *****
32 32 *****
33 33 001266 017501 000002 000002 000002 000002 000002 000002 000002 000002
34 34 001272 012737 010420 172416 010420 172416 010420 172416 010420 172416
35 35 001300 042701 004400 004400 004400 004400 004400 004400 004400 004400
36 36 001304 010137 172416 172416 172416 172416 172416 172416 172416 172416
37 37 001310 105737 172414 172414 172414 172414 172414 172414 172414 172414
38 38 001314 100375 100375 100375 100375 100375 100375 100375 100375 100375
39 39 001316 105037 172414 172414 172414 172414 172414 172414 172414 172414
40 40 001322 000207 000207 000207 000207 000207 000207 000207 000207 000207
41 41
42 42

```

;WHICH SUBROUTINE
 ;DISABLE LIU INTERRUPTS
 ;ENABLE LIU
 ;ENCODE BYTE COUNT
 ;DECODE BYTE COUNT
 ;CALL SWITCH(X) -SET OR RESET LINE SWITCHES
 ;04=SET PRIMARY LINE
 ;32=RESET PRIMARY LINE
 ;08=SET BACKUP LINE
 ;64=RESET BACKUP LINE
 SWITCH: MOV @ARG1(R5),R1
 MOV #4368.,@OPREG
 ADD #2304.,R1
 MOV R1,@OPREG
 TSTB @BCSR
 BPL .-4
 CLRB @BCSR
 RTS PC
 ;SW SET
 ;WCR: MODSTAT
 ;RD
 ;VALID WRITE
 ;NO LOOP UNTIL READY
 ;CLEAR DONE BIT

```
1 *****
2 ***** PROCEDURE STATUS
3 *****
4 *****
5
6
7
8
9
10 STATUS: MOV      @ARG1(R5), R1
11          CMP      #1, R1
12          BEQ      2$
13          MOV      #4352., @#0PREG
14          MOV      #1280., @#0PREG
15          MOV      @#10BUF, R0
16          BIC      #177400, R0
17          MOV      R0, @ARG2(R5)
18          RTS
19          MOV      #4480., @#0PREG
20          MOV      #1280., @#0PREG
21          MOV      @#10BUF, R0
22          BIC      #177400, R0
23          MOV      R0, @ARG2(R5)
24          RTS
25          MOV      #4352., @#0PREG
26          MOV      #1280., @#0PREG
27          MOV      @#10BUF, R1
28          BIC      #177400, R1
29          BITB     #BIT03, R1
30          BEQ      3$
31          MOV      #1, @ARG1(R5)
32          RTS
33          MOV      #0, @ARG1(R5)
34          RTS
35
36
```

***** PROCEDURE STATUS

CALL STATUS(X,DATA) -READ STATUS BYTES 0/1
;
;X=0 STATUS BYTE 0
;X=1 STATUS BYTE 1

WHICH BYTE
;COMPARE

WCR : RS(0)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA

WCR : RS(1)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA

STATUS BYTE 1
;RS
;FETCH DATA
;BUFFER FULL
;LOAD RETURN
;LOAD RETURN

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****

*****
***** PROCEDURE WRITE TOKEN (WTKEN) *****
*****
;CALL WTKEN -LOADS AND WRITES A TOKEN TO LOOP

WTKEN: MOV #4480., @#OPREG
        MOV #1280., @#OPREG
        MOV @#IOBUF, R1
        BITB @BIT03, R1
        BEQ WTKG
        MOV #4368., @#OPREG
        MOV #2323., @#OPREG
        TSTB @#CSR
        BPL .-4
        CLRB @#CSR
        RTS PC
WTKG: MOV #4576., @#OPREG
        MOV #WTD, @#BAR
        MOV #2, R2
        NEG R2
        MOV R2, @#WCR
        MOV #10240., @#OPREG
        NOP
        TSTB @#CSR
        NOP
        CLRB @#CSR
        MOV #4368., @#OPREG
        MOV #2322., @#OPREG
        TSTB @#CSR
        BPL .-4
        CLRB @#CSR
        RTS PC
WTD: .BYTE 0,377
        377
        000
        35
        36
        37
        38
        39
        40
        41
        42
        43
        44
        45
        46
        47
        48

;READ STATUS(1)
;RS
;FETCH DATA
;BUFFER LOADED
;NO GENERATE TOKEN
;MODSTAT
;MARK OB0F/OB1F/WTCMD
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT

;SELECT BUFFER 1
;ADDRESS OF WRITE TOKEN
;% OF BYTES
;FORMAT IT
;DMA BYTE COUNT
;FIRE DMA
;DELAY
;GOOD DMA
;ERROR IF HERE
;CLEAR DONE
;MODSTAT
;MARK OB1F/WTCMD
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;RETURN
;A WTKEN

;TIME+1
;?TIME FULL
;CLEAR CLOCK

```


FIM.MACRO
NODE 23

MACRO V03.01 4-JUN-79 17:50:23 PAGE 7

```

1 *****
2 *****
3 ***** PROCEDURE ACRAM *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 001676 017500 000002 000000 000000 000000
15 001702 022700 000000 000000 000000 000000
16 001706 001035 000000 000000 000000 000000
17 001710 017500 000004 000000 000000 000000
18 001714 017501 000006 000000 000000 000000
19 001720 012737 010402 172416 172416 172416
20 001726 062700 004400 000000 000000 000000
21 001732 010037 172416 172416 172416 172416
22 001736 105737 172414 172414 172414 172414
23 001742 100375 172414 172414 172414 172414
24 001744 105037 172414 172414 172414 172414
25 001750 012737 010401 172416 172416 172416
26 001756 062701 004400 000000 000000 000000
27 001762 010137 172416 172416 172416 172416
28 001766 105737 172414 172414 172414 172414
29 001772 100375 172414 172414 172414 172414
30 001774 105037 172414 172414 172414 172414
31 002000 000207 172414 172414 172414 172414
32 002002 017500 000004 000000 000000 000000
33 002006 012737 010402 172416 172416 172416
34 002014 062700 004400 000000 000000 000000
35 002020 010037 172416 172416 172416 172416
36 002024 105737 172414 172414 172414 172414
37 002030 100375 172414 172414 172414 172414
38 002032 105037 172414 172414 172414 172414
39 002036 012737 010401 172416 172416 172416
40 002044 012737 001400 172416 172416 172416
41 002052 105737 172414 172414 172414 172414
42 002056 100375 172414 172414 172414 172414
43 002060 105037 172414 172414 172414 172414
44 002064 013700 172416 172416 172416 172416
45 002070 042700 177760 177760 177760 177760
46 002074 010075 000006 000006 000006 000006
47 002100 000207 000207 000207 000207 000207
48
49

```

; RAM
 ; WHICH OPERATION
 ; READ OF
 ; ADDRESS
 ; WRITE DATA
 ; SEL LDACR
 ; WD/DATA (ADDR)
 ; WRITE DATA
 ; VALID WRITE
 ; NO LOOP UNTIL READY
 ; CLEAR DONE BIT
 ; SEL ACRAM
 ; WD/DATA (CMD)
 ; WRITE
 ; VALID WRITE
 ; NO LOOP UNTIL READY
 ; CLEAR DONE BIT
 ; ADDRESS TO READ
 ; SEL LDACR
 ; WD/DATA (ADDR)
 ; WRITE DATA
 ; VALID WRITE
 ; NO LOOP UNTIL READY
 ; CLEAR DONE BIT
 ; SEL ACRAM
 ; READ DATA
 ; VALID READ
 ; NO LOOP UNTIL READY
 ; CLEAR DONE BIT
 ; FETCH DATA
 ; CLEAR BITS
 ; RTN DATA

; CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
 ; CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
 ;
 ;0110 =NREAD
 ;0100 =DREAD
 ;0111 =NULL
 ;0000 =WIDEN

RAM:
 MOV @ARG1(R5), R0
 CMP #0, R0
 BNE RDRAM
 MOV @ARG2(R5), R0
 MOV @ARG3(R5), R1
 MOV #4354., @OPREG
 ADD #2304., R0
 MOV R0, @OPREG
 TSTB @CSR
 BPL .-4
 CLRB @CSR
 MOV #4353., @OPREG
 ADD #2304., R1
 MOV R1, @OPREG
 TSTB @CSR
 BPL .-4
 CLRB @CSR
 RTS PC
 MOV @ARG2(R5), R0
 MOV #4354., @OPREG
 ADD #2304., R0
 MOV R0, @OPREG
 TSTB @CSR
 BPL .-4
 CLRB @CSR
 MOV #4353., @OPREG
 ADD #2304., @OPREG
 TSTB @CSR
 BPL .-4
 CLRB @CSR
 MOV @IDBUF, R0
 BIC #177760, R0
 MOV R0, @ARG3(R5)
 RTS PC


```

1 1
2 2
3 3
4 4
5 5
6 002102 017502 000002 000002 000002 000002 000002 000002 000002 000002
7 002106 012700 000340 000340 000340 000340 000340 000340 000340 000340
8 002112 106400 000000 000000 000000 000000 000000 000000 000000 000000
9 002114 012737 000000 000000 000000 000000 000000 000000 000000 000000
10 002122 005402 000000 000000 000000 000000 000000 000000 000000 000000
11 002124 010237 000000 000000 000000 000000 000000 000000 000000 000000
12 002130 012737 000000 000000 000000 000000 000000 000000 000000 000000
13 002136 012737 000000 000000 000000 000000 000000 000000 000000 000000
14 002144 000240 000000 000000 000000 000000 000000 000000 000000 000000
15 002146 005737 000000 000000 000000 000000 000000 000000 000000 000000
16 002152 000240 000000 000000 000000 000000 000000 000000 000000 000000
17 002154 005037 000000 000000 000000 000000 000000 000000 000000 000000
18 002160 012737 000000 000000 000000 000000 000000 000000 000000 000000
19 002166 012737 000000 000000 000000 000000 000000 000000 000000 000000
20 002174 012702 000000 000000 000000 000000 000000 000000 000000 000000
21 002200 005402 000000 000000 000000 000000 000000 000000 000000 000000
22 002202 010237 000000 000000 000000 000000 000000 000000 000000 000000
23 002206 012737 000000 000000 000000 000000 000000 000000 000000 000000
24 002214 000240 000000 000000 000000 000000 000000 000000 000000 000000
25 002216 005737 000000 000000 000000 000000 000000 000000 000000 000000
26 002222 000240 000000 000000 000000 000000 000000 000000 000000 000000
27 002224 005037 000000 000000 000000 000000 000000 000000 000000 000000
28 002230 012737 000000 000000 000000 000000 000000 000000 000000 000000
29 002236 012737 000000 000000 000000 000000 000000 000000 000000 000000
30 002244 005737 000000 000000 000000 000000 000000 000000 000000 000000
31 002250 000375 000000 000000 000000 000000 000000 000000 000000 000000
32 002252 005037 000000 000000 000000 000000 000000 000000 000000 000000
33 002256 012700 000000 000000 000000 000000 000000 000000 000000 000000
34 002262 006400 000000 000000 000000 000000 000000 000000 000000 000000
35 002264 000207 000000 000000 000000 000000 000000 000000 000000 000000
36
37
38

```

```
1 5 002266 017500 000002
2 6 002272 022700 000000
3 7 002276 001004
4 8 002300 012700 010410
9 9 002304 000167 000046
10 10 002310 022700 000001
11 11 002314 001004
12 12 002316 012700 010610
13 13 002322 000167 000030
14 14 002326 022700 000002
15 15 002332 001004
16 16 002334 012700 010510
17 17 002340 000167 000012
18 18 002344 022700 000004
19 19 002350 001020
20 20 002352 012700 010710
21 21 002356 010037 172416
22 22 002362 012737 001400
23 23 002370 105737 172414
24 24 002374 100375
25 25 002376 013700 172416
26 26 002402 042700 177400
27 27 002406 010075 000004
28 28 002412 000207

*****
***** PROCEDURE READ BUFFER POINTER *****
*****
RDPNT:  MOV @ARG1(R5), R0
        CMP  R0, R0
        BNE  1$
        MOV  @4360., R0
        JMP  4$
        CMP  R0, R0
        BNE  2$
        MOV  @4488., R0
        JMP  4$
        CMP  R0, R0
        BNE  3$
        MOV  @4424., R0
        JMP  4$
        CMP  R0, R0
        BNE  5$
        MOV  @4552., R0
        MOV  R0, @0DFREG
        MOV  @768., @0DFREG
        TSTB @CSR
        BFL  .-4
        MOV  @0IDRUF, R0
        BIC  @177400, R0
        MOV  R0, @ARG2(R5)
        RTS  PC

1$:
2$:
3$:
4$:
5$:

;FETCH COMMAND
;INBUFO ?
;NO
;RDBUFADR INO
;INBUFI ?
;NO
;RDBUFADR INI
;OUTBUFO ?
;NO
;RDBUFADR OUTO
;OUTBUFI ?
;NO RETURN
;RDBUFADR OUTI
;RDBUFADR
;RD
;READY
;NO LOOP UNTIL
;FETCH POINTER
;CLEAR MST
;RETURN POINTER
```

FDM. MACRO
NODE 23

MACRO V03.01 4-JUN-79 17:50:23 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 002414 017500 000002          ;CALL RSTART(0) -CAUSES A SOFTWARE HALT
10 002420 022700 000000          ;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
11 002424 001002          ;CALL RSTART(2) -LOAD MODE(173000)
12 002426 000000          *****
13 002430 000207          *****
14 002432 022700          *****
15 002436 001005          *****
16 002440 004767 175432          ;MODE
17 002444 012700 000040          ;LOAD ADDRESS
18 002450 000110          *****
19 002452 022700 000002          ;RESTART PROGRAM
20 002456 001002          ;INT LIU FIRST
21 002460 000177 000002          ;START ADDRESS
22 002464 000207          ;LOAD MODE
23 002466 173000          *****
24
25
26

```

```

1 1 *****.ENABLE LSB*****
2 2 *****
3 3 *****PROCEDURE SIG INTERFACE*****
4 4 *****
5 5 *****
6 6 SIG::      MOV      R0,      -(SP)      ;SAVE REGISTERS
7 7             MOV      R1,      -(SP)
8 8             MOV      R2,      -(SP)
9 9             MOV      $XINRF, R0      ;BUFFER ADDRESS
10 10            MOV      $40., R1      ;BYTE COUNT
11 11            TSTB     @#HRCSCR      ;PORT READY
12 12            BPL      .-4          ;LOOP UNTIL
13 13            MOVB     @#HREUF, R2    ;FETCH BYTE
14 14            BIC      $177400, R2   ;CLEAR MST
15 15            MOV     R2, 10FLG      ;ITS THE BYTE COUNT
16 16            MOVB     R2, (R0)+
17 17            DEC      R1            ;COUNT-1
18 18            TSTB     @#HRCSCR      ;PORT READY
19 19            BPL      .-4          ;LOOP UNTIL
20 20            MOVB     @#HREUF, R2    ;FETCH BYTE
21 21            BIC      $177400, R2   ;CLEAR MST
22 22            MOV     R2, (R0)+
23 23            DEC      R1            ;STORE IT
24 24            BNE      2$
25 25            BIC      $100, @#HRCSCR ;READ 39 BYTES
26 26            MOV     (SP)+, R2      ;NO MORE UNTIL FORTRAN
27 27            MOV     (SP)+, R1      ;RESTORE REGISTERS
28 28            MOV     (SP)+, R0
29 29            RTI
30 30
31 31
32 32          ;SEND OVER TO SIG
33 33          SOUT::    MOV      @ARG1(R5),R0      ;BYTE COUNT
34 34                  TSTB     @#HXCSCR          ;READY TO SNED
35 35                  BPL      .-4              ;LOOP UNTIL
36 36                  MOVB     R0, @#HXBUF        ;SEND IT
37 37                  RTS
38 38
39 39          ;REPORT NODE READY
40 40
41 41          READY::   TSTB     @#HXCSCR          ;PORT READY
42 42                  BPL      .-4              ;LOOP UNTIL
43 43                  MOVB     $122, @#HXBUF      ;SEND R
44 44                  BIS      $100, @#HRCSCR     ;ENABLE INTERRUPTS
45 45                  RTS
46 46                  PC
47 47
48 48

```

FDM, MACRO
NODE 23

MACRO V03.01 4-JUN-79 17:50:23 PAGE 12

```

1
2
3
4
5 000000
6
7 000000
8 000400
9 001000
10 001002
11 001004
12 001006
13
14
15 000000
16
17 000000
18 000400
19 001000
20 001400
21 002000
22 002002
23 002004
24
25
26
27

*****
**** COMMON DATA AREA *****
*****
.PSECT DFM,RW,D,GBL,REL,OVR

OUTBF: .BLKB 256.
INBF: .BLKB 256.
IWRTTH: .BLKW
IWRT: .BLKW
STAT: .BLKW
LTIME: .BLKW
.EVEN

.PSECT BUFS,RW,D,GBL,REL,OVR

XOUTBF: .BLKB 256.
XINBF: .BLKB 256.
ACKSEQ: .BLKB 256.
LIDFD: .BLKB 256.
IOFLG: .BLKW
LLFLG: .BLKW
IRSEND: .BLKW
.EVEN

.END
000001

```


FDM-MACRO
SYMBOL TABLE
MACRO V03.01 4-JUN-79 17:50:23 PAGE 12-1

ACKSER	001000R	003	B1\$	000722R	E1\$	001236R	MASTER	000000RG	SOUT	002576RG	002
AREA	001206R		B2\$	000510R	G4\$	002444R	NDXX =	000000	STAT	001004R	
ARG1 =	000002		B3\$	000534R	HREUF =	177562	ND24 =	000001	STAT	001422RG	
ARG2 =	000004		B4\$	000560R	HRC5R =	177560	NHOST =	000000	STATUS	001324RG	
ARG3 =	000006		B5\$	000606R	HXBUR =	177566	OFREG =	172416	STAT\$	000404R	
ARG4 =	000010		B6\$	000624R	HXC5R =	177564	OUTBF	000000R	002 SWITCH	001266RG	
BAR	= 172410		7\$	000570R	INBF	000400R	002 RAM	001676RG	TIME	001654RG	
B1T00 =	000001		CAUSE	001204R	IOBUF =	172416	KDFNT	002266RG	WCR =	172412	
B1T01 =	000002		CRCOK	001054R	IOFLG	002000R	003 KDRAM	002002R	WTD	001652R	
B1T02 =	000004		CSR	= 172414	IRSEND	002004R	003 READY	002616RG	WTKG	001552R	
B1T03 =	000010		DATA	001212R	IWRT	001002R	002 KDM	002466R	WTKEN	001474RG	
B1T04 =	000020		DESTR	001256RG	IWRTH	001000R	002 RST	002464R	WTRAM	001710R	
B1T05 =	000040		DHOST =	000001	LIDFD	001400R	003 RSTART	002414RG	XCIO =	000001	
B1T06 =	000100		DMAOK	001160R	LIO	000430RG	RS0	002420R	XGIO =	000001	
B1T07 =	000200		EMBF	001024R	LIUINT	000076RG	RS1	002432R	XINEF	000400R	003
B1T08 =	000400		ENABLE	001214RG	LLFLG	002002R	RS2	002452R	XOUTBF	000000R	003
B1T14 =	040000		END\$	001202R	LPINPT=	***** G	003 RS2	001000R	XGIO =	000000	
B1T15 =	100000		ENSTR	001246RG	LPOUT	002102RG	RTI\$	001000R	ZEROP	000304R	
B0\$	000644R		E0\$	001226R	LTIME	001006R	SIO	002470RG			

* ABS. 000000 000
002642 001
DFM 001010 002
RUF5 002006 003
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 51 PAGES
,DK:FDM=DK:FDM

PAGE 001

12:00:00

01-01-79

COM 23.01M

PROGRAM:

MCCMPTOR

FOR NODAL/NOI INE
FOR NODAL/NOI INE
FOR DISC/NOI INE
MAC FOR

PAGE 001

12:00:00

01-JUL-79

LNK23.COM

Program:

decwriter

ASSIGN DX0: DN:

R LINK

DX1:NDRE-3,DX1:MAF-DX1:NDUAL /C/L/W

DX1:FIM/C

DX1:NDUAL I/C

DX1:DS-DC//

\$STHR

ASSIGN DX1: DN:

1.5 NODE 24 DMCP (DBMS)

The Data Base Management Service (DBMS) performs the data base maintenance functions. It maintains the display files for the human interface User Language, system configuration files, equipment status files, and object files for the various modules. The User Language runs on the DBMS and is used to control the system. Mini disks are used to store the various files. The DBMS may also be used as a Program Development Unit (PDU) with a local CRT terminal when not being used for the MSCDM application. The PDU develops and maintains software for the MSCDM. The PDU runs the RT-11 Operating System and it may be used as a general purpose processor.

1.5.1 Program Descriptions

1.5.1.1 Refer to Section 1.1 for descriptions of routines - NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME, RAM, LPOUT, RDPNT, RSTART.

1.5.1.2 Subroutine USRLANG

MSCDM user language is the applications program for node 24, refer to Section 1.10 for description and listings.

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:41:40      PAGE 001

0001      PROGRAM NODAL
0002      INTEGER*2 XINQ,PXINQ
0003      INTEGER*2 PINQ,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBNF
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      INTEGER*2 ST,DEVNUM,FTYPE,RECNUM
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,TEMP(6),OUTBF,INBF
0008      LOGICAL*1 F7,F8,F9,F10,F11,F12,IBUFF
0009      LOGICAL*1 PACK,ICODE,ETX,CR,LF,MONTOR,ISLID,ICON,DUM,IBUF
0010      INTEGER*2 LTIME,ACKTIM,NEWTIM,OLDTIM,IWRITM,TIMLIM,ATIMLM
0011      INTEGER*2 KEYREC
0012      COMMON /MESS/ MESSEQ
0013      COMMON /DFM/ OUTBF(256),INBF(256),
1          IWRITM,IWRT,STAT,LTIME
0014      COMMON/QUE/ XINQ(15),PXINQ(2),INQ(15),PINQ(2)
0015      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
1          LIDFD(256),IOFLG,LLFLG,IRSEND
0016      COMMON /FRE/ FREE(30),IFR,IFRSZ
0017      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0018      COMMON /PACK/ PACK(256,30)
0019      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0020      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0021      COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0022      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
1          KEYREC,IBUF(80)
0023      COMMON /M3000/ IFUNT,ITYPE,INODE,IMEAS
0024      COMMON /M4000/ FTYPE,RECNUM,F7,F8,F9,F10,F11,F12
0025      COMMON /M6000/ IDISP,IEQUIP,ICHAN,INUM,IBUFF(20),IFLAG,ICNT
0026      CALL MASTER(0)
0027      CALL INIT(0)
0028      CALL ENABLE(1)
0029      IFIRST=0
0030      CALL USRLNG(0)
0031      5 IF(IABORT.EQ. 1) GOTO 999
0033      IF(IFLAG.EQ. 0 .OR. IRSEND.EQ. 1) GOTO 25
0035      CALL M6000
0036      DO 27 J=1,20000
0037      CONTINUE
0038      25 IF(PINQ(1).LT. PINQ(2)) GOTO 40
0040      CALL ENABLE(0)
0041      Q2=DEQUE(PINQ,INQ,1)
0042      CALL ENABLE(1)
0043      CALL DESTR(PACK(254,Q2),LEN)
0044      DO 26 I=1,80
0045      ICODE(I)=0
0046      CONTINUE
0047      DO 30 I=1,LEN
0048      ICODE(I)=PACK(I,Q2)
0049      CONTINUE
0050      CALL USRLNG(LEN)
0051      IFR=IFR+1
0052      FREE(IFR)=Q2
0053      40 CALL STATB0(IS)

```



```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:41:40      PAGE 002

0054      IF (IS .EQ. 1) GOTO 100
0056      IF (LFLG .GT. 0) CALL LINLOS
0058      80      IF (OUTFCT .EQ. 1) GOTO 100
0060      IF ((IFULL .EQ. 1) .OR. (ISENT .EQ. 1)) GOTO 100
0062      IF (IRSEND .EQ. 1) GOTO 85
0064      IF (PXING(1) .LT. PXING(2)) GOTO 100
0066      CALL ENABLE(0)
0067      Q1=DEQUE(PXING,XING,1)
0068      CALL ENABLE(1)
0069      OUTQ=Q1
0070      85      CALL DEST(PACK(254,OUTQ),Q2)
0071      IF (Q2 .LT. 6) GOTO 100
0073      DO 90 I=1,Q2
0074          OUTBF(I)=PACK(I,OUTQ)
0075      90      CONTINUE
0076      OUTBF(Q2+1)=0
0077      IPT=OUTBF(5)
0078      OUTBF(Q2+2)=LIDFD(IPT)
0079      IF (Q2+2 .LT. 6 .OR. Q2+2 .GT. 255) GOTO 100
0081      IFULL=1
0082      ISENT=1
0083      CALL LPOUT(Q2+2)
0084      IRSEND=1
0085      ACKTIM=0
0086      IWRTM=0
0087      IWRT=1
0088      INFCT=0
0089      OLDTIM=NEWTIM
0090      NEWTIM=LTIME
0091      IF (IWRT .EQ. 0) GOTO 120
0093      IWRTM=IWRTM+(NEWTIM-OLDTIM)
0094      IF (IWRTM .LT. TIMLM) GOTO 120
0096      CALL WTOKEN
0097      IWRTM=0
0098      IWRT=0
0099      120      IF (ISENT .EQ. 0) GOTO 130
0101      ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0102      IF (ACKTIM .LT. ATIMLM) GOTO 130
0104      CALL ENABLE(0)
0105      CALL ACKNAK(0)
0106      CALL ENABLE(1)
0107      CONTINUE
0108      130      GOTO 5
0109      999      CALL ENABLE(0)
0110      CALL MASTER(1)
0111      STOP
0112      END

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:42:02      PAGE 001

0001      FUNCTION IGETSP(N)
0002      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003      INTEGER*2 FREE,FLWCNT
0004      INTEGER*2 OUTFCT,OUTQ,RESNLM
0005      COMMON /FRE/ FREE(30),IFR,IFRSZ
0006      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0007      IF (IFR .LT. 1) CALL INIT(1)
0008      IGETSP=FREE(IFR)
0009      IFR=IFR-1
0010      RETURN
0011      END
0012

```

PAGE 001

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:42:14

```

0001 SUBROUTINE ENQUE(A,B,N)
0002 LOGICAL*1 ETX,CR,LF,MONTR,ISLID,DUM
0003 INTEGER*2 XING,PXING,INQ,PING
0004 INTEGER*2 FLWCNT,OUTQ,A(2),B(15)
0005 INTEGER*2 RESNLM,OUTFCT
0006 COMMON /QUE/ XING(15),PXING(2),INQ(15),PING(2)
0007 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,IQLNTH,OUTFCT,IFULL,OUTQ,
      1 RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0008 10 IQHEAD=A(1)
0009 IQTAIL=A(2)
0010 IF(IQTAIL.EQ. 1) GOTO 20
0011 IQTAIL=IQTAIL-1
0012 B(IQTAIL)=N
0013 A(2)=IQTAIL
0014 GOTO 999
0015
0016 20 IF(IQHEAD.GE. IQLNTH) GOTO 40
0017 NN=IQHEAD-IQTAIL
0018 DO 30 I=1,NN+1
0019 30 B(IQLNTH+1-I)=B(IQHEAD+1-I)
0020 A(1)=IQLNTH
0021 A(2)=IQLNTH-NN
0022 GOTO 10
0023
0024 40 CALL INIT(1)
0025 999 RETURN
0026 END

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:42:27      PAGE 001

0001      FUNCTION DEQUE(A,B,ID)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOR
0003      INTEGER*2 XING,PXING,INQ,PING
0004      INTEGER*2 FLWCNT,A(2),B(15),DEQUE,OUTFCT,OUTQ,RESNLM
0005      COMMON/QUE/ XING(15),PXING(2),INQ(15),PING(2)
0006      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,IQLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0007      IQHEAD=A(1)
0008      DEQUE=B(IQHEAD)
0009      IF(ID.NE. 1) GOTO 999
0010      IF(IQHEAD.NE. 0) GOTO 10
0011      A(1)=IQLNTH
0012      A(2)=IQLNTH+1
0013      GOTO 999
0014      10 A(1)=IQHEAD-1
0015      999 RETURN
0016      END
0017
0018

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:42:56      PAGE 001

0001      SUBROUTINE ACKNAK(N)
0002      INTEGER*2 FLWCNT,OUTFCT,STAT,XINQ,PXINQ
0003      INTEGER*2 PINQ,FREE,OUTQ,FINDPK
0004      INTEGER*2 T1,T2,T3,T4,T5,RESNLM
0005      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0006      LOGICAL*1 PACK,OUTBF,INBF,LCOMT(40)
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0008      INTEGER*2 LTIME,IWRTH
0009      REAL*8 COMT(5)
0010      EQUIVALENCE(COMT,LCOMT)
0011      DATA COMT(1) / 'MESSAGE ', COMT(2) / 'NOT RECE', COMT(3) / 'IVED FRO',
1      COMT(4) / 'M NODE ', COMT(5) / 'TO NODE ' /
0012      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
1      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0013      COMMON /DFM/ OUTBF(256),INBF(256),
1      IWRTH,IWRT,STAT,LTIME
0014      COMMON      PACK(256,30)
0015      COMMON/QUE/ XINQ(15),PXINQ(2),INQ(15),PINQ(2)
0016      COMMON /FRE/ FREE(30),IFR,IFRSZ
0017      COMMON /MESS/ MESSEQ
0018      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1      IOFLG,LLFLG,IRSEND
0019      IF (N.NE. 1) GOTO 10
0020      IFR=IFR+1
0021      DO 5 I=1,256
0022      PACK(I,OUTQ)=0
0023      5      CONTINUE
0024      IFULL=0
0025      ISENT=0
0026      IRSEND=0
0027      FREE(IFR)=OUTQ
0028      GOTO 999
0029      10      T2=PACK(256,OUTQ)
0030      T2=T2+1
0031      IF(T2.LE. RESNLM) GOTO 120
0032      T3=PACK(3,OUTQ)
0033      T4=MOD(T3,64)
0034      IF(T4.LT. 32) GOTO 110
0035      T5=IGETSP(N)
0036      IF(MESSEQ.EQ. 126) MESSEQ=0
0037      MESSEQ=MESSEQ+1
0038      PACK(1,T5)=0
0039      PACK(2,T5)=MESSEQ
0040      PACK(3,T5)=0
0041      PACK(4,T5)=0
0042      PACK(5,T5)=MONTOR
0043      PACK(6,T5)=ISLID
0044      DO 20 I=7,60
0045      PACK(I,T5)=*40
0046      20      CONTINUE
0047      DO 30 I=7,9
0048      PACK(I,T5)=LF
0049      30      CONTINUE
0050      DO 40 I=1,8
0051      40      CONTINUE
0052      40      CONTINUE
0053      DO 40 I=1,8
0054      40      CONTINUE

```


PAGE 002

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:42:56

```

0055      PACK(I+9,TS)=LCONT(I)
0056      CONTINUE
0057      DO 50 I=9,16
0058      PACK(I+9,TS)=LCONT(I)
0059      CONTINUE
0060      DO 60 I=17,24
0061      PACK(I+9,TS)=LCONT(I)
0062      CONTINUE
0063      DO 70 I=25,31
0064      PACK(I+9,TS)=LCONT(I)
0065      CONTINUE
0066      ENCODE(3,80,PACK(41,TS))PACK(6,OUTQ)
0067      FORMAT(I3)
0068      DO 90 I=33,40
0069      PACK(I+12,TS)=LCONT(I)
0070      CONTINUE
0071      ENCODE(3,80,PACK(53,TS))PACK(5,OUTQ)
0072      PACK(56,TS)=LF
0073      PACK(57,TS)=LF
0074      PACK(58,TS)=CR
0075      PACK(59,TS)=ETX
0076      CALL ENSTR(PACK(254,TS),59)
0077      CALL ENQUE(PXINQ,XINQ,TS)
0078      DO 102 I=1,256
0079      PACK(I,OUTQ)=0
0080      CONTINUE
0081      ISENT=0
0082      IRSEND=0
0083      IFULL=0
0084      IWRT=0
0085      IFR=IFR+1
0086      FREE(IFR)=OUTQ
0087      GOTO 999
0088      110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089      PACK(256,OUTQ)=0
0090      ISENT=0
0091      IFULL=0
0092      IF(IRSEND.EQ.1) GOTO 999
0093      CALL ENQUE(PXINQ,XINQ,OUTQ)
0094      GOTO 999
0095      120 PACK(256,OUTQ)=T2
0096      IFULL=0
0097      ISENT=0
0098      IF(IRSEND.EQ.1) GOTO 999
0099      CALL ENQUE(PXINQ,XINQ,OUTQ)
0100      RETURN
0101      999
0102      END
0103

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:43:17      PAGE 001

0001      SUBROUTINE INPTQ(L)
0002      INTEGER*2 STAT,XINQ,PXINQ
0003      INTEGER*2 PINQ,FLWCNT,OUTFCT,T1,RESNLM,OUTQ
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,TEMP,OUTBF,INBF
0005      LOGICAL*1 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
0006      INTEGER*2 LTIME,IWRTIM
0007      COMMON /DFM/ OUTBF(256),INBF(256),
0008      1      IWRTIM,IWRT,STAT,LTIME
0009      1      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010      1      LIDFD(256),IOFLG,LLFLG,IRSEND
0011      1      COMMON
0012      1      COMMON /QUE/ XINQ(15),PXINQ(2),INQ(15),PINQ(2)
0013      1      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
0014      1      RESNLM,IATRI,ISLID,MONITOR,LF,ETX,CR,DUM
0015      IF(L.LT. 1) GOTO 30
0016      IF(L.GT. 256) GOTO 30
0017      LI=L-5
0018      II=INBF(6)
0019      T1=IGETSP(N)
0020      DO 15 I=1,LI
0021      15      PACK(I,T1)=INBF(I)
0022      CALL ENSTR(PACK(254,T1),LI)
0023      CALL ENQUE(PINQ,INQ,T1)
0024      N=2
0025      GOTO 40
0026      N=1
0027      GOTO 40
0028      N=0
0029      DO 50 I=1,2
0030      50      OUTBF(I)=INBF(I)
0031      CONTINUE
0032      NN=1
0033      IF(N.EQ. 0) NN=128
0034      OUTBF(3)=NN
0035      OUTBF(4)=0
0036      OUTBF(5)=INBF(6)
0037      OUTBF(6)=ISLID
0038      OUTBF(7)=ETX
0039      OUTBF(8)=0
0040      OUTBF(9)=LIDFD(INBF(6))
0041      CALL LPOUT(9)
0042      IF(N.NE. 2) GOTO 999
0043      ACKSEQ(II)=INBF(2)
0044      RETURN
0045      999      END
0046
0047
0048

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:43:34      PAGE 001

0001      SUBROUTINE LPINPT(LI)
0002      INTEGER*2 STAT,FLWCNT,OUTFCT,OUTQ,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONTR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRTTH
0006      COMMON /DFM/ OUTBF(256),INBF(256),IWRTM,IWRT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1      RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0009      CC1=INBF(3)
0010      CC2=INBF(4)
0011      IF(CC1 .LT. 0) GOTO 25
0012      IF(CC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0013      CC3=MOD(CC1,2)
0014      CC4=MOD(CC1,256)
0015      IF ((CC3 .GE. 1 .OR. CC4 .GE. 128) .AND. ISENT .EQ. 1) GOTO 20
0016      IF ((INBF(1) .EQ. 85) .AND. INBF(2) .EQ. 170) GOTO 99
0017      CC5=MOD(INBF(3),64)
0018      IF(CC5 .GE. 32) CALL INPTQ(LI)
0019      GOTO 99
0020      20      CC6=MOD(CC1,64)
0021      IF(CC6 .GE. 32) IALTRT=1
0022      CC7=MOD(CC1,2)
0023      IF(CC7 .GE. 1) GOTO 30
0024      25      CALL ACKNAK(0)
0025      GOTO 99
0026      30      CALL ACKNAK(1)
0027      GOTO 99
0028      40      CALL INPTQ(LI)
0029      99      CONTINUE
0030      RETURN
0031      END
0032
0033
0034
0035
0036
0037

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:43:49      PAGE 001

0001      SUBROUTINE INIT(IDDM)
0002      INTEGER*2 ST,DEVNUM,FTYPE,RECNUM
0003      INTEGER*2 XINQ,PXINQ,PINQ,FREE
0004      INTEGER*2 STAT,FLWCNT,OUTFCT
0005      INTEGER*2 SETPRM,RSTPRM,SETBKP,OUTQ,RESNLM,RSTBKP
0006      LOGICAL*1 ICON,ICON,DUM,IBUFF,IBUF
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0008      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID
0009      LOGICAL*1 F7,F8,F9,F10,F11,F12
0010      INTEGER*2 LTIME,OLDTIM,TIMLIM,ACKTIM,ATIMLM,IWRTTM
0011      COMMON /DFM/ OUTBF(256),INBF(256),
1          IWRTTM,IWRT,STAT,LTIME
0012      COMMON/QUE/ XINQ(15),PXINQ(2),INQ(15),PINQ(2)
0013      COMMON /MESS/ MESSEQ
0014      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1          IOFLG,LLFLG,IRSEND
0015      COMMON /FRE/ FREE(30),IFR,IFRSZ
0016      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0017      COMMON
0018      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0019      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0020      COMMON /USER/ ST(4,9),IND,ICD(256),NRCNO,NOREC,ICON(256)
0021      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
2          KEYREC,IBUF(80)
0022      COMMON /M3000/ IFUNT,ITYPE,INODE,IMEAS
0023      COMMON /M4000/ FTYPE,RECNUM,F7,F8,F9,F10,F11,F12
0024      COMMON /M6000/ IDISP,IEQUIP,ICHAN,INUM,IBUFF(20),IFLAG,ICNT

0025      CALL RAM(0,5,4)
0026      CALL RAM(0,255,0)
0027      MESSEQ=0
0028      IABORT=0
0029      IFLAG=0
0030      IOFLG=0
0031      IRSEND=0
0032      LLFLG=0
0033      IFRSZ=30
0034      INFCT=0
0035      ISLID=24
0036      CR='015'
0037      LF='012'
0038      ETX='003'
0039      RESNLM=2
0040      IOLNTH=15
0041      IALTRT=0
0042      FLWCNT=0
0043      IWRTTM=0
0044      TIMLIM=50
0045      SETPRM=4
0046      RSTPRM=32
0047      SETBKP=8
0048      RSTBKP=64
0049      ATIMLM=200

```

```

0050 MONITOR=25
0051 IWRT=0
0052 OUTFACT=0
0053 IFULL=0
0054 ISENT=0
0055 DO 40 I=1,20
0056   LIDFD(I)=4
0057   LIDFD(21)=1
0058   LIDFD(22)=3
0059   LIDFD(23)=6
0060   LIDFD(24)=5
0061   LIDFD(25)=7
0062   LIDFD(26)=8
0063   LIDFD(27)=9
0064   LIDFD(28)=2
0065   DO 50 I=29,39
0066     LIDFD(I)=0
0067     DO 60 I=40,44
0068       LIDFD(I)=1
0069       DO 70 I=45,59
0070         LIDFD(I)=0
0071         DO 80 I=60,64
0072         LIDFD(I)=2
0073         DO 90 I=65,79
0074         LIDFD(I)=0
0075         DO 100 I=80,84
0076         LIDFD(I)=4
0077         DO 110 I=85,99
0078         LIDFD(I)=0
0079         DO 120 I=100,104
0080         LIDFD(I)=5
0081         DO 130 I=105,256
0082         LIDFD(I)=0
0083         PING(1)=IQLNTH
0084         PING(2)=IQLNTH+1
0085         PXING(1)=IQLNTH
0086         PXING(2)=IQLNTH+1
0087         DO 10 I=1,IFRSZ
0088           PACK(255,I)=0
0089           PACK(256,I)=0
0090           DO 20 I=1,IFRSZ
0091             FREE(I)=1
0092             DO 30 I=1,256
0093               ACKSEQ(I)=256
0094               IFR=IFRSZ
0095               IF(IIDM.EQ.1) RETURN
0096               CALL ASSIGN(1,'TT:')
0097               CALL ASSIGN(2,'STATUS.DAT')
0098               DEFINE FILE 2(1016,10,U,J6)
0099               READ(2,1)(ICODE(I),I=1,20)
0100               CALL ASSIGN(3,'CDIR.DAT')
0101               DEFINE FILE 3(10,40,U,J1)
0102               READ(3,1)(ICODE(I),I=1,1,80)
0103               CALL ASSIGN(4,'TDIR.DAT')

```


PAGE 003

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:43:49

```
0104    DEFINE FILE 4(10,40,U,J2)
0105    READ(4,1)(ICODE(I),I=1,80)
0106    CALL ASSIGN(5,'CATD.DAT')
0107    DEFINE FILE 5(100,40,U,J3)
0108    READ(5,1)(ICODE(I),I=1,80)
0109    CALL ASSIGN(6,'TKND.DAT')
0110    DEFINE FILE 6(100,40,U,J4)
0111    READ(6,1)(ICODE(I),I=1,80)
0112    CALL ASSIGN(7,'REPORT.DAT')
0113    DEFINE FILE 7(10,40,U,J5)
0114    READ(7,1)(ICODE(I),I=1,80)
0115    CALL ASSIGN(8,'MSG.DAT')
0116    DEFINE FILE 8(400,40,U,NRCND)
0117    RETURN
0118    END
```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:44:14      PAGE 001

0001      SUBROUTINE LINLOS
0002      REAL*8 RM1(5),LIN08,LIN18
0003      INTEGER*2 XING,PXING,PING
0004      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM,T1
0005      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,XOUTBF,XINBF,DUM
0006      LOGICAL*1 PACK,M1(40),LIN0(8),LIN1(8),ACKSEQ,LIDFD
0007      COMMON/QUE/ XING(15),PXING(2),INQ(15),FINQ(2)
0008      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0010      COMMON /MESS/ MESSEQ
0011      COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0012      1      IOFLG,LLFLG,IRSEND
0013      DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0014      DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0015      DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0016      DATA RM1(5)/'DE 24 '//
0017      EQUIVALENCE(LIN0,LIN08)
0018      EQUIVALENCE(LIN1,LIN18)
0019      IS=LLFLG
0020      CALL ENABLE(0)
0021      T1=IGETSP(N)
0022      CALL ENABLE(1)
0023      PACK(1,T1)=0
0024      IF(MESSEQ.EQ.126) MESSEQ=0
0025      MESSEQ=MESSEQ+1
0026      PACK(2,T1)=MESSEQ
0027      PACK(3,T1)=0
0028      PACK(4,T1)=0
0029      PACK(5,T1)=25
0030      PACK(6,T1)=24
0031      DO 20 I=7,9
0032      PACK(I,T1)=LF
0033      CONTINUE
0034      DO 30 I=1,22
0035      PACK(I+9,T1)=M1(I)
0036      IF(15.EQ.1) GO TO 50
0037      DO 40 I=1,8
0038      PACK(I+31,T1)=LIN0(I)
0039      GO TO 70
0040      DO 60 I=1,8
0041      PACK(I+31,T1)=LIN1(I)
0042      DO 80 I=23,40
0043      PACK(I+17,T1)=M1(I)
0044      PACK(58,T1)=CR
0045      PACK(59,T1)=LF
0046      PACK(60,T1)=ETX
0047      CALL ENSTR(PACK(254,T1),60)
0048      CALL ENABLE(0)
0049      CALL ENQUE(PXING,XING,T1)
0050      CALL ENABLE(1)
0051      LLFLG=0
0052      RETURN
0053
0054

```

PAGE 002

U02.1-1 Tue 05-Jun-79 13:44:14

FDRTRAN IV

0055 END

```

FDM.MACRO      MACRO V03.01 5-JUN-79 13:44:43 PAGE 1

1  .TITLE FDM.MACRO
2  .IDENT /V3.0/
3  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
4  .GLOBL WTONEN,RAM,STATUS,STATBO
5  .GLOBL LPOUT,RSTART,RDPNT,DESTR,ENSTR,LPINPT
6  .NLIST CND
7  .PSECT
8
9
10 000000
11 000001
12 000002
13 000003
14 000004
15 000005
16 000006
17 000007
18
19
20
21 172410
22 172412
23 172414
24 172416
25 172416
26 177560
27 177562
28 177564
29 177566
30
31
32
33 000001
34 000001
35 000001
36
37
38
39 000002
40 000004
41 000006
42 000010
43
44 100000
45 040000
46 000400
47 000200
48 000100
49 000040
50 000020
51 000010
52 000004
53 000002
54 000001
55

      R0 =Z0
      R1 =Z1
      R2 =Z2
      R3 =Z3
      R4 =Z4
      R5 =Z5
      SP =Z6
      PC =Z7

; INTERFACE ADDRESS
      BAR= 172410
      WCR= 172412
      CSR= 172414
      IDBUF= 172416
      OPREG= 172416
      HKCSR= 177560
      HRBUF= 177562
      HXCSR= 177564
      HXBUF= 177566

; INTERFACE OPTIONS
      XCIO=1
      XSIO=1
      XGIO=1

; LOCAL VARIABLES
      ARG1= 2
      ARG2= 4
      ARG3= 6
      ARG4= 10

      BIT15= 100000
      BIT14= 40000
      BIT08= 400
      BIT07= 200
      BIT06= 100
      BIT05= 40
      BIT04= 20
      BIT03= 10
      BIT02= 4
      BIT01= 2
      BIT00= 1

```

```

1 *****
2 ***** PROCEDURE MASTER START UP ***** *
3 *****
4 *****
5 MASTER: MOV @ARG1(R5), R0 ;START OR RESTORE
6 CMP #1, R0 ;RESTORE?
7 BEQ RSTVEC ;YES JUMP OVER START
8 MOV #340, R0 ;PRI=7
9 MTPS R0
10 MOV #124, R0 ;LIU HANDLER
11 MOV #110, (R0)+ ;PRI=7
12 MOV #340, (R0) ;CLOCK VECTOR
13 MOV #100, R0 ;SAVE IT FOR RECOVERY
14 MOV (R0), VECSAV ;CLOCK
15 MOV #01670, ;SAVE IT FOR RECOVERY
16 BIC #40100, @#CSR ;DISABLE INTERRUPTS
17 JSR PC, LIUINT ;CLEAR LIU
18 CLR @#CSR ;CLEAR BLUI CSR
19 MOV #000, R0 ;PRI=0
20 MTPS R0 ;CPU DOWN
21 RTS FC
22 RSTVEC: MOV #100, R0 ;VECTOR SPOT
23 MOV VECSAV, (R0) ;RESTORE IT
24 RTS FC
25 VECSAV: .WORD 0 ;VECTOR MAILBOX,
26
27 *****
28 ***** PROCEDURE INITIALIZE ***** *
29 *****
30 *****
31 ;CLEAR ACRAM
32 LIUINT: MOV #4354., @#OPREG ;LDADR
33 MOV #2304., @#OPREG ;ADDRESS=0
34 TSTB @#CSR ;GOOD WD
35 BPL -4 ;NO RETRY
36 CLRB @#CSR ;SEL ACRAM
37 MOV #4353., @#OPREG ;COUNT
38 MOV #-256., R0 ;WRITE A NULL
39 MOV #2311., @#OPREG ;GOOD WD
40 TSTB @#CSR ;NO LOOP UNTIL READY
41 BPL -4 ;COUNT+1
42 INC R0
43 BNE 1$
44
45 ;CLEAR INPUT/OUTPUT BUFFERS
46 MOV #10410, R1 ;RDBUFADR CMD
47 MOV #10440, R0 ;SEL INBUFO
48 JSR PC, ZERORF ;SET POINTER=0
49 MOV #1400, @#OPREG ;FALSE READ DATA
50 TSTB @#CSR ;GOOD READ
51 BPL -4 ;NO RETRY

```


FDM-MACRO MACRO 003.01 5-JUN-79 13:44:43 PAGE 2-1

```

76 000226 105037 172414 CLR B
77 000232 012701 010610 MOV
78 000236 012700 010640 MOV
79 000242 004767 000052 JSR
80 000246 012737 001400 MOV
81 000254 105737 172414 TST B
82 000260 100375 BFL
83 000262 105037 CLR B
84 000266 012701 172414 MOV
85 000272 012700 010540 MOV
86 000276 004767 000016 JSR
87 000302 012701 010710 MOV
88 000306 012700 010740 MOV
89 000312 004767 000002 JSR
90 000316 000440 BR
91 000320 010137 172416 MOV
92 000324 012737 001400 MOV
93 000332 105737 172414 TST B
94 000336 100375 BFL
95 000340 105037 CLR B
96 000344 013702 172414 MOV
97 000350 042702 177400 BIC
98 000354 010037 172416 MOV
99 000360 022702 000000 CMP
100 000364 001412 BEQ
101 000366 012737 001400 MOV
102 000374 105737 172414 TST B
103 000400 100375 BFL
104 000402 105037 CLR B
105 000406 005302 DEC
106 000410 000763 BR
107 000412 010037 172416 MOV
108 000416 000207 RTS
109
110
111
112 000420 012737 010400 MOV
113 000426 012737 002400 MOV
114 000434 012737 002400 MOV
115 000442 000207 RTS
116
117

;CLEAR DONE BIT
;RDBUFADR COMMAND
;SEL INBUF1 COMMAND
;POINTER=0
;FALSE READ DATA
;GOOD RD
;NO RETRY
;CLEAR DONE BIT
;RDBUFADR COMMAND
;SEL OUTBUFO COMMAND
;RDBUFADR COMMAND
;SEL OUTBUFO COMMAND
;ZERO BUFFER POINTER
;GO CLEAR STATUS
;RDBUFADR
;RD
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;FETCH POINTER
;CLEAR MST BYTE
;SEL BUFFER
;POINTER=0
;YES RECHECK
;NO FALSE RD
;GOOD RD
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;LOCAL POINTER-1
;BR UNTIL 0
;SEL BUFFER
;RETURN

;WCR:RS
;RS(FALSE)
;RS

;READ AND CLEAR STATUS
;STAT$:
MOV 4352., @00FREG
MOV 1280., @00FREG
MOV 1280., @00FREG
RTS PC

```

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 000444 010046 LIO: MOV R0, -(SF)
7 000446 010146 MOV R1, -(SF)
8 000450 010246 MOV R2, -(SF)
9 000452 010346 MOV R3, -(SF)
10 000454 010446 MOV R4, -(SF)
11 000456 010546 MOV R5, -(SF)
12 000460 005737 TST @CSR
13 000464 100133 RPL RTI$
14 000466 042737 BIC $BIT14, @CSR
15 000474 012737 MOV 172414, @OFREG
16 000502 012737 MOV 172416, @OFREG
17 000510 013767 MOV 172416, @IDRUF, CAUSE
18 000516 042767 BIC 177400, CAUSE
19
20
21
22 000524 132767 000004 BITB $BIT02, CAUSE
23 000532 001406 BEQ B3$
24 000534 012702 MOV $4360, R2
25 000540 012703 MOV $4384, R3
26 000544 004767 JSR PC, EMBF
27 000550 132767 000042 BITB $BIT03, CAUSE
28 000556 001406 BEQ B4$
29 000560 012702 MOV $4488, R2
30 000564 012703 MOV $4512, R3
31 000570 004767 JSR PC, EMBF
32 000574 132767 000016 BITB $BIT04, CAUSE
33 000602 001407 BEQ B5$
34 000604 132767 000200 BITB $BIT07, CAUSE
35 000612 001403 BEQ B6$
36 000614 012767 000002 MOV $2, LLFLG
37 000622 132767 000040 BITB $BIT05, CAUSE
38 000630 001403 BEQ B6$
39 000632 012767 000001 MOV $1, LLFLG
40 000640 132767 000100 BITB $BIT06, CAUSE
41 000646 001404 BEQ B6$
42 000650 005067 CLR IWRTH
43 000654 005067 CLR IWR
44
45
46
47 000660 132767 000001 BITB $BIT00, CAUSE
48 000666 001423 BEQ B1$
49 000670 012701 MOV $INEF, R1
50 000674 012702 MOV $4360, R2
51 000700 012703 MOV $4384, R3
52 000704 012704 MOV $BIT00, R4
53 000710 004767 JSR PC, EMBF
54 000714 012705 001222 MOV $AREA, R5
55 000720 012767 000001 MOV $1, AREA
56 000726 010267 MOV 000274, DATA
57 000732 004767 JSR PC, LPINPT

```

```

;SAVE R0
;SAVE R1
;SAVE R2
;SAVE R3
;SAVE R4
;SAVE R5
;? LIU INTERRUPTED
;NO
;YES/DISABLE LIU
;WCR : RS(0)
;RD
;CLEAR UNUSED BITS

```

```

;PERFORM OPERATION DEPENDING ON BITS

```

```

;OV-FL
;RDRUFADR INO
;SEL INBUFO
;GO EMPTY
;OV-FL
;RDRUFADR IN1
;SEL INBUF1
;GO EMPTY
;LINE-LOSS PRIMARY
;LATCH ON
;YES LOOP AROUND
;FLAG PRIMARY SW. FAIL
;BACKUP LINE LOSS
;FLAG BACKUP SW. FAIL
;WRITE TOKEN DETECT
;CLEAR TOKEN VAR

```

```

;INBUFO FULL
;NO
;BUFFER ADDRESS
;RDRUFADR COMM
;SEL BUF
;CRC BIT
;EMPTY BUFFER
;DATA LINK AREA
;1 VARIABLE
;CRC OR BYTE COUNT
;CALL FORTRAN QUE'ER

```

FDM.MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 3-1

```

58 000736 132767 000002 000254 BIT1: CAUSE
59 000744 001423 REG RTI$
60 000746 012701 000400' INBF, R1
61 000752 012702 010610 MOV $488., R2
62 000756 012703 010640 MOV $512., R3
63 000762 012704 000002 MOV BIT01, R4
64 000766 04767 000046 PC, EMBF
65 000772 012705 001222' AREA, R5
66 000776 012767 000001 MOV $1, AREA
67 001004 010267 000216 MOV R2, DATA
68 001010 004767 000000G PC, LFINFT
69
70 001014 012605 RTI$:
71 001016 012604 MOV (SP)+, R4
72 001020 012603 MOV (SP)+, R3
73 001022 012602 MOV (SP)+, R2
74 001024 012601 MOV (SP)+, R1
75 001026 012600 MOV (SP)+, R0
76 001030 052737 040000 BIT14, @CSR
77 001036 000002 RIS
78 RTI
79 001040 012737 010600 MOV $480., @OPREG
80 001046 012737 002400 MOV $1280., @OPREG
81 001054 013700 172416 MOV @IOBUF, R0
82 001060 130400 BITB R4, R0
83 001062 001002 BNE CRCOK
84 001064 012704 177777 MOV $-1, R4
85 001070 010237 172416 MOV R2, @OPREG
86 001074 012737 001400 MOV $768., @OPREG
87 001102 105737 172414 TSTB @CSR
88 001106 100375 BPL -4
89 001110 013702 172416 MOV @IOBUF, R2
90 001114 042702 177400 MOV $177400, R2
91 001120 010200 BIC R2, R0
92 001122 005400 NEG R0
93 001124 010037 172412 MOV R0, @WCR
94 001130 010137 172410 MOV R1, @RAR
95 001134 010337 172416 MOV R3, @OPREG
96 001140 012737 001400 MOV $768., @OPREG
97 001146 105737 172414 TSTB @CSR
98 001152 100375 BPL -4
99 001154 012737 021000 MOV $B704., @OPREG
100 001162 000240 MOV @CSR
101 001164 105737 172414 TSTB DMAOK
102 001170 100401 BHI
103 001172 000240 MOV NOF
104 001174 012737 004400 MOV $2304., @OPREG
105 001202 105737 172414 TSTB @CSR
106 001206 100375 BPL -4
107 001210 005704 TST R4
108 001212 100001 BPL END$
109 001214 010402 MOV R4, R2
110 001216 000207 RTS PC
111 001220 000000 CAUSE: .WORD 0
112 001222 000000 AREA: .WORD 0
113 001224 001226' DATA: .WORD 0
114 001226 000000

```

FDM.MACRO

MACRO V03.01 5-JUN-79 13:44:43 PAGE 4

```

1      .ENABLE LSH
2      *****
3      ;**** PROCEDURE ENABLE ****
4      *****
5
6      001230    017501    000002
7      001234    022701    000001
8      001240    001404
9      001242    042737    040000    172414
10     001250    002037    PC
11     001252    052737    040000    172414
12     001260    000207    PC
13
14
15     ;FORTHAN ENCODE/DECODE MACRO
16
17     001262    017575    000004    000002
18     001270    000207    RTS
19     001272    017575    000002    000004
20     001300    000207    RTS
21
22
23
24
25
26
27
28
29
30
31
32
33     001302    017501    000002
34     001306    012737    010420    172416
35     001314    062701    004400
36     001320    010137    172416
37     001324    105737    172414
38     001330    100375    BFL
39     001332    105037    CLR#B
40     001336    000207    RTS
41
42

```


FIM.MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 5

[illegible]


```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****

***** PROCEDURE ACRAM *****
*****
;CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;0100 =DREAD
;0111 =NULL
;0000 =WOKEN

RAM:      MOV     @RG1(R5), R0
          CMP     #0, R0
          BNE     KDRAM
          WTRAM:  MOV     @RG2(R5), R0
          MOV     @RG3(R5), R1
          MOV     #4354., @#OPREG
          ADD     #2304., R0
          MOV     R0, @#OPREG
          TSTB    @#CSR
          BPL     .-4
          CLRB    @#CSR
          MOV     #4353., @#DPREG
          ADD     #2304., R1
          MOV     R1, @#OPREG
          TSTB    @#CSR
          BPL     .-4
          CLRB    @#CSR
          RTS     PC
          KDRAM:  MOV     @RG2(R5), R0
          MOV     #4354., @#OPREG
          ADD     #2304., R0
          MOV     R0, @#OPREG
          TSTB    @#CSR
          BPL     .-4
          CLRB    @#CSR
          MOV     #4353., @#OPREG
          MOV     #768., @#DPREG
          TSTB    @#CSR
          BPL     .-4
          CLRB    @#CSR
          MOV     @#IOBUF, R0
          BIC     #17760, R0
          MOV     R0, @RG3(R5)
          RTS     PC

; RAM
; WHICH OPERATION
; READ OP
; ADDRESS
; WRITE DATA
; SEL LDATA
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; WD/DATA (CMD)
; WRITE
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; ADDRESS TO READ
; SEL LDATA
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; READ DATA
; VALID READ
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; FETCH DATA
; CLEAR BITS
; RTN DATA

```

FDM.MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 8

```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38
*****
***** PROCEDURE WRITE LOOP (LFOUT)
*****
LFOUT: MOV @ARG1(R5), R2
MOV #340, R0
MTFS
MOV @OUTBF, @BBAR
NEG R2
MOV R2, @WCR
MOV #4448, @OPREG
MOV #10240, @OPREG
NOP
TSTB @CSR
NOP
CLR B
MOV #4576, @OPREG
MOV @WTD, @BBAR
MOV #2, R2
NEG R2
MOV R2, @WCR
MOV #10240, @OPREG
NOP
TSTB @CSR
NOP
CLR B
MOV #4368, @OPREG
MOV #2307, @OPREG
TSTB @CSR
BPL -4
CLRB
MOV #000, R0
MTFS
RTS
*****
; BYTE COUNT
; PRI=7
; BUS ADDRESS
; 2'S COMP COUNT
; COUNT
; OR0 COMMAND
; DMA GO
; INTERFACE TIME
; DMA OK
; ERROR IF HERE
; OR1 COMMAND
; ADDRESS OF WRITE TOKEN
; BYTE COUNT
; FORMAT FOR BLUI
; DMA BYTE COUNT
; FIRE DMA
; DELAY
; GOOD DMA
; ERROR IF NOT
; CLEAR DONE
; MODSTAT
; BUFFERS FULL
; GOOD WRITE
; NO LOOP UNTIL
; CLEAR DONE BIT
; PRI=0
; FLOWER CPU

```

```
*****  
***** PROCEDURE READ BUFFER POINTER *****
```

FDM.MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 002430 017500 000002 RSTART: MOV BARG1(RS), R0 ;MODE
10 002434 022700 000000 RSO: #0, R0 ;LOAD ADDRESS
11 002440 001002 BNE RSI
12 002442 000000 HALT
13 002444 000207 RTS PC
14 002446 022700 000001 RS1: #1, R0 ;RESTART PROGRAM
15 002452 001005 BNE RS2 PC, LIUINT ;INT LIU FIRST
16 002454 004767 175432 JSR PC, LIUINT ;START ADDRESS
17 002460 012700 000040 G4: #40, R0 ;LOAD MODE
18 002464 000110 JMP (R0)
19 002466 022700 000002 RS2: #2, R0
20 002472 001002 BNE RSI
21 002474 000177 000002 JMP BROM
22 002500 000207 RST: RTS PC
23 002502 173000 ROM: .WORD 173000
24
25
26
235

```


FDM.MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 11

```

1 *****
2 *****
3 *****
4 *****
5 000000 .PSECT IFM,RW,D,GBL,REL,OVR
6
7 000000 OUTBF: .BLKB 256.
8 000400 INBF: .BLKB 256.
9 001000 IWRTM: .BLKW
10 001002 IWRT: .BLKW
11 001004 STAT: .BLKW
12 001006 LTIME: .BLKW
13 .EVEN
14
15 000000 .PSECT BUFS,RW,D,GBL,REL,OVR
16
17 000000 XOUTBF: .BLKB 256.
18 000400 XINBF: .BLKB 256.
19 001000 ACKSEQ: .BLKB 256.
20 001400 LIDFD: .BLKB 256.
21 002000 IOFLG: .BLKW
22 002002 LLFLO: .BLKW
23 002004 IRSEND: .BLKW
24 .EVEN
25
26
27 000001 .END

```

 ***** COMMON DATA AREA *****

FDM-MACRO MACRO V03.01 5-JUN-79 13:44:43 PAGE 11-1

SYMBOL TABLE

ACKSEQ	001000R	003 B0\$	000660R	E0\$	001242R	LFOUT	002116RG	STATB0	001436RG	
AREA	001222R	B1\$	000736R	E1\$	001252R	LTIME	001006R	STATUS	001340RG	
ARG1	= 000002	B2\$	000524R	G4\$	002460R	MASTER	000000RG	STAT\$	000420R	
ARG2	= 000004	B3\$	000550R	HRBUF	= 177562	OPREG	= 172416	SWITCH	001302RG	
ARG3	= 000006	B4\$	000574R	HRCR\$	= 177560	OUTBF	000000R	TIME	001670RG	
ARG4	= 000010	B5\$	000622R	HXBUR	= 177566	RAM	001712RG	VECSAV	000110R	
BAR	= 172410	B6\$	000640R	HXC\$R	= 177564	RDPNT	002302RG	WCR	= 172412	
BIT00	= 000001	B7\$	000660R	INBF	000400R	002 RDRAM	002016R	WTD	001666R	
BIT01	= 000002	CAUSE	001220R	IOBUF	= 172416	ROM	002502R	WTKG	001566R	
BIT02	= 000004	CRCOK	001070R	IOFLG	002000R	003 RST	002500R	WTOKEN	001510RG	
BIT03	= 000010	CSR	= 172414	IRSEND	002004R	003 RSTART	002430RG	WTRAM	001724R	
BIT04	= 000020	DATA	001226R	IURT	001002R	002 RSTVEC	000076R	XCIO	= 000001	
BIT05	= 000040	DESTR	001272RG	IURTTM	001000R	002 RS0	002434R	XGIO	= 000001	
BIT06	= 000100	DHACK	001174R	LIDFD	001400R	003 RS1	002446R	XINBF	000400R	003
BIT07	= 000200	ENBF	001040R	LID	000444RG	RS2	002466R	XOUTBF	000000R	003
BIT08	= 000400	ENABLE	001230RG	LIUINT	000112RG	RTI\$	001014R	XSIO	= 000001	
BIT14	= 040000	END\$	001216R	LLFLG	002002R	003 STAT	001004R	002 ZEROBP	000320R	
BIT15	= 100000	ENSTR	001262RG	LPINPT	= ***** G					

. ABS. 000000 000
 002504 001
 DFM 001010 002
 BUFS 002006 003
 ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 51 PAGES
 ,DK:FDM=DK:FDM

PAGE 001

12:00:00

01-III-79

CONFIDENTIAL

PROGRAM:

DESCRIPTION:

FOR NOBIL/UNITES:2/NOI IRE
FOR NOBIL 1/NOI IRE
MAC EUM

W. L. LARSEN

0-1114-1

371 1001-2011

THE JOURNAL OF THE

07/1:0:0000/H:180
07/1:0:0000/D:180

2017:06:01

0X1:0/000001X0
0X1:0/000001X0

1941: 1942

0.05 : 0.01 : 0.01 :

1.6 NODE 25 (OCRI)

The Operator Control and Report Interface (OCRI) performs the functions of interfacing to the site operator. The User Language interface, when the OCRI terminal is ATTACHED to the DMBS node, allows the operator to control the site, request status information concerning site and equipment performance and prepare site reports which are forwarded to other sites such as the ACOC. Operator-to-operator messages are also supported by the user Language. The OCRI terminal can be used to print event reports, fault reports, and alarms for the simulated equipment generated by the FIAC and SDCA modules, and error messages concerning the FDM itself generated by the nodal modules (e.g., Loop-Back in effect, queue overflow).

1.6.1 Program Descriptions

1.6.1.1 Refer to Section 1.1 for descriptions of routines-NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME, RAM, LPOUT, RDPNT, RSTART.

1.6.1.2 Subroutine HINPUT (FORTRAN)

This subroutine is called by the interrupt handler of the LA-36. This routine is where the ATTACH command is checked and decoded. If the message is not a command it is queued.

1.6.1.3 Subroutine CIO (MACRO)

This subroutine is the interrupt handler for the LA-36 terminal. It reads the ASCII data as it's typed, checks for deletes, control, and xon-xoff characters and stores them until a RETURN is typed. When RETURN is detected subroutine HINPUT is called.

1.6.1.4 Subroutine COUT (MACRO)

This subroutine is used to print a message received from the loop on the LA-36. It's passed one argument the byte count to be printed.

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:30:22      PAGE 001

0001      PROGRAM NODAL
0002      INTEGER*2 XING,XOUTQ,ACKQ,PXING,FXOUTQ,PACKQ
0003      INTEGER*2 PING,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP,IXON
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDF,OUTBF,INBF
0007      LOGICAL*1 PACK,ETX,CR,LF,MONITOR,ISLID,DUM
0008      INTEGER*2 LTIME,ACNTIM,NEWTIM,OLDTIM,IKRTIM,ATIMLM,ATIMLM
0009      COMMON /MESS/ MESSEQ
0010      COMMON /DFM/ OUTBF(256),INBF(256),IKRTIM,IMRT,STAT,LTIME
0011      COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),FXOUTQ(2),
0012      1      ACKQ(16),PACKQ(2),INQ(16),PING(2)
0013      1      LIDF(256),IOFLG,LLFLG,IRSEND
0014      COMMON /FRE/ FREE(64),IFR,IFRSZ
0015      COMMON /TIM/ OLDTIM,ATIMLM,ACKTIM,ATIMLM
0016      1      PACK(256,64)
0017      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0018      1      RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0019      COMMON /CRT/IXON
0020      IXON=0
0021      CALL MASTER
0022      CALL INIT
0023      CALL ENABLE(1)
0024      5      CONTINUE
0025      25      IF(PING(1).LT. PING(2)) GOTO 40
0026      CALL ENABLE(0)
0027      Q2=DEQUE(PING,INQ,1)
0028      CALL ENABLE(1)
0029      CALL DESTR(PACK(254,Q2),LEN)
0030      DO 30 I=1,LEN
0031      XOUTBF(I)=PACK(I,Q2)
0032      30      CONTINUE
0033      ITERM=XOUTBF(6)
0034      IF(ITERMX.EQ. 27 .AND. IXON.EQ. 1) GOTO 31
0035      IF(ITERMX.EQ. 28 .AND. IXON.EQ. 1) GOTO 31
0036      CALL COUT(LEN)
0037      31      IFR=IFR+1
0038      FREE(IFR)=Q2
0039      40      CALL STATB0(1S)
0040      IF(1S.EQ. 1) GOTO 100
0041      IF(10FLG.GT. 0) CALL HINPUI
0042      IF(11FLG.GT. 0) CALL LINLOS
0043      IF(12FLG.GT. 0) CALL LINLOS
0044      IF(13FLG.GT. 0) CALL LINLOS
0045      IF(14FLG.GT. 0) CALL LINLOS
0046      IF(15FLG.GT. 0) CALL LINLOS
0047      IF(16FLG.GT. 0) CALL LINLOS
0048      IF(17FLG.GT. 0) CALL LINLOS
0049      IF(18FLG.GT. 0) CALL LINLOS
0050      IF(19FLG.GT. 0) CALL LINLOS
0051      IF(20FLG.GT. 0) CALL LINLOS
0052      IF(21FLG.GT. 0) CALL LINLOS
0053      IF(22FLG.GT. 0) CALL LINLOS
0054      IF(23FLG.GT. 0) CALL LINLOS
0055      IF(24FLG.GT. 0) CALL LINLOS
0056      IF(25FLG.GT. 0) CALL LINLOS
0057      IF(26FLG.GT. 0) CALL LINLOS
0058      IF(27FLG.GT. 0) CALL LINLOS
0059      IF(28FLG.GT. 0) CALL LINLOS
0060      IF(29FLG.GT. 0) CALL LINLOS
0061      IF(30FLG.GT. 0) CALL LINLOS

```

PAGE 002

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:30:22

```

0062      OUTBF(I)=PACK(I,OUTQ)
0063      CONTINUE
0064      OUTBF(Q2+1)=0
0065      IPT=OUTBF(5)
0066      OUTBF(Q2+2)=LIDF(IPT)
0067      IF(Q2+2.LT.6.OR.Q2+2.GT.255) GOTO 100
0068      IFULL=1
0069      ISENT=1
0070      CALL LPOUT(Q2+2)
0071      IRSEND=1
0072      ACKTIM=0
0073      IWRTIM=0
0074      IWRT=1
0075      INFCT=0
0076      100 OLDTIM=NEWTIM
0077      110 NEWTIM=LTIME
0078      IF(IWRT.EQ.0) GOTO 120
0079      IWRTIM=IWRTIM+(NEWTIM-OLDTIM)
0080      IF(IWRTIM.LT.TIMLM) GOTO 120
0081      CALL WTOKEN
0082      IWRTIM=0
0083      IWRT=0
0084      120 IF(ISENT.EQ.0) GOTO 130
0085      ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0086      IF(ACKTIM.LT.ATIMLM) GOTO 130
0087      CALL ENABLE(0)
0088      CALL ACKNAK(0)
0089      CALL ENABLE(1)
0090      130 CONTINUE
0091      GOTO 5
0092      END
0093
0094
0095
0096
0097

```

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:30:43 PAGE 001

```
0001      FUNCTION IGTSF(N)
0002      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003      INTEGER*2 FREE,FLWCNT
0004      INTEGER*2 OUTFCT,OUTQ,RESNLM
0005      COMMON /FRE/ FREE(64),IFR,IFRSZ
0006      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0007      1      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0008      IF (IFR .LT. 1) CALL INIT
0009      IGTSF=FREE(IFR)
0010      IFR=IFR-1
0011      RETURN
0012      END
```

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:30:54 PAGE 001

```

0001 SUBROUTINE ENQUE(A,B,N)
0002 LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003 INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWCNT,OUTQ,A(2),B(16)
0005 INTEGER*2 RESNLM,OUTFCT
0006 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007      1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0008      1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1 RESNLM,IALTRI,ISLID,MONITOR,LF,ETX,CR,DUM
0010      10 IQHEAD=A(1)
0011      IQTAIL=A(2)
0012      IF(IQTAIL.EQ.1) GOTO 20
0013      IQTAIL=IQTAIL-1
0014      B(IQTAIL)=N
0015      A(2)=IQTAIL
0016      GOTO 999
0017      20 IF(IQHEAD.GE.IQLNTH) GOTO 40
0018      NN=IQHEAD-IQTAIL
0019      DO 30 I=1,NN+1
0020      30 B(IQLNTH+1-I)=B(IQHEAD+1-I)
0021      A(1)=IQLNTH
0022      A(2)=IQLNTH-NN
0023      GOTO 10
0024      40 CALL INIT
0025      999 RETURN
0026      END

```



```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:31:08      PAGE 001

0001      FUNCTION DEQUE(A,B,ID)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOR
0003      INTEGER*2 XINQ,XING,XOUTQ,PACKQ,INQ,PINQ
0004      INTEGER*2 FLWCNT,A(2),B(16),DEQUE,OUTFCT,OUTQ,RESNLM
0005      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      1      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0007      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0009      IQHEAD=A(1)
0010      DEQUE=B(IQHEAD)
0011      IF(ID.NE.1) GOTO 999
0012      IF(IQHEAD.NE.0) GOTO 10
0013      A(1)=IQLNTH
0014      A(2)=IQLNTH+1
0015      GOTO 999
0016      10      A(1)=IQHEAD-1
0017      999      RETURN
0018      END

```

171

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:31:46      PAGE 001

0001      SUBROUTINE ACKNAK(N)
0002      INTEGER*2 FLWCNT,OUTFCT,STAT,XINQ,FXINQ,FXOUTQ
0003      INTEGER*2 ACKQ,PACKQ,PINQ,FREE,OUTQ,XOUTQ
0004      INTEGER*2 T1,T2,T3,T4,T5,RESNLM
0005      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0006      LOGICAL*1 PACK,OUTBF,INBF,LCOMT(40)
0007      INTEGER*2 LTIME,IWRTH
0008      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0009      REAL*8 COMT(5)
0010      EQUIVALENCE(COMT,LCOMT)
0011      DATA COMT(1)/'MESSAGE '//,COMT(2)/'NOT SENT',//,COMT(3)/' FROM '//,
1      COMT(4)/'NODE '//,COMT(5)/' TO NODE'//
0012      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1      RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0013      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
1      LIDFD(256),IDFLG,LLFLG,IRSEND
0014      COMMON /DFM/OUTBF(256),INBF(256),IWRTH,IWRT,STAT,LTIME
0015      COMMON /PACK/ PACK(256,64)
0016      COMMON /QUE/ XINQ(16),FXINQ(2),XOUTQ(16),FXOUTQ(2),ACKQ(16),
1      COMMON /FREE/ FREE(64),IFR,IFRSZ
0017      COMMON /MESS/ MESSEQ
0018      IF (N.NE. 1) GOTO 10
0019      IFR=IFR+1
0020      DO 5 I=1,256
0021      PACK(I,OUTQ)=0
0022      5 CONTINUE
0023      IFULL=0
0024      ISENT=0
0025      FREE(IFR)=OUTQ
0026      IRSEND=0
0027      GOTO 999
0028      10 T2=PACK(256,OUTQ)
0029      T2=T2+1
0030      IF(T2.LE. RESNLM) GOJP 120
0031      T3=PACK(3,OUTQ)
0032      T4=MOD(T3,64)
0033      IF(T4.LT. 32) GOTO 110
0034      DO 20 I=7,60
0035      XOUTBF(I)=*40
0036      20 CONTINUE
0037      DO 30 I=7,9
0038      XOUTBF(I)=LF
0039      30 CONTINUE
0040      DO 40 I=1,8
0041      XOUTBF(I+9)=LCOMT(I)
0042      40 CONTINUE
0043      DO 50 I=9,16
0044      XOUTBF(I+9)=LCOMT(I)
0045      50 CONTINUE
0046      DO 60 I=17,22
0047      XOUTBF(I+9)=LCOMT(I)
0048      60 CONTINUE
0049      DO 70 I=23,29
0050      XOUTBF(I+9)=LCOMT(I)
0051      70 CONTINUE
0052      DO 80 I=30,36
0053      XOUTBF(I+9)=LCOMT(I)
0054      80 CONTINUE

```

PAGE 002

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:31:46

```

0054 XOUTBF(I+7)=LCOMT(I)
0055 CONTINUE
0056 ENCODE(3,80,XOUTBF(38)) ISLID
0057 FORMAT(I3)
0058 DO 90 I=33,40
0059 XOUTBF(I+11)=LCOMT(I)
0060 CONTINUE
0061 ENCODE(3,80,XOUTBF(52)) MONITOR
0062 XOUTBF(55)=LF
0063 XOUTBF(56)=LF
0064 XOUTBF(57)=CR
0065 XOUTBF(58)=ETX
0066 CALL COUT(58)
0067 DO 102 I=1,556
0068 PACK(I,OUTQ)=0
0069 CONTINUE
0070 ISENT=0
0071 IFULL=0
0072 IWR=0
0073 IFR=IFR+1
0074 FREE(IFR)=OUTQ
0075 IRSEND=0
0076 GOTO 999
0077 110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0078 PACK(256,OUTQ)=0
0079 ISENT=0
0080 IFULL=0
0081 IF(IRSEND.EQ.1) GOTO 999
0083 CALL ENQUE(PXING,XING,OUTQ)
0084 GOTO 999
0085 120 PACK(256,OUTQ)=T2
0086 ISENT=0
0087 IFULL=0
0088 IF(IRSEND.EQ.1) GOTO 999
0090 CALL ENQUE(PXING,XING,OUTQ)
0091 RETURN
0092 END

```

PAGE 001

Tue 05-Jun-79 13:32:06

002.1-1

```

SUBROUTINE INPTQ(L)
  INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ
  INTEGER*2 /ACKQ,PINQ,FLWCNT,OUTFCT,T1,RESNLM,OUTQ
  LOGICAL*41 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
  LOGICAL*41 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
  INTEGER*2 LTIME,IWRTTM,IXON
  COMMON /DFM/ OUTBF(256),INBF(256),IWRTTM,IWRT,STAT,LTIME
  COMMON /BFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
    LIDFD(256),IOFLG,LLFLG,IRSEND
  COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
    ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
  COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
    RESNLM,IALTST,ISLID,MONITOR,LF,ETX,CR,DUM
  COMMON /CRT/IXON
  IF(L.LT. 1) GOTO 30
  IF(L.GT. 256) GOTO 30
  LI=L-2
  II=INBF(6)
  IF(II.EQ. 27 .AND. IXON.EQ. 1) GOTO 20
  IF(II.EQ. 28 .AND. IXON.EQ. 1) GOTO 20
  IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
  T1=IGETSP(N)
  DO 10 I=1,LI
    10 PACK(I,T1)=INBF(I)
    CALL ENSTR(PACK(254,T1),LI)
    CALL ENQUE(PINQ,INQ,T1)
  N=2
  GOTO 40
  20 N=1
  GOTO 40
  30 N=0
  40 DO 50 I=1,2
    OUTBF(I)=INBF(I)
  50 CONTINUE
  NN=1
  IF(N.EQ. 0) NN=128
  OUTBF(3)=NN
  OUTBF(4)=0
  OUTBF(5)=INBF(6)
  OUTBF(6)=ISLID
  OUTBF(7)=ETX
  OUTBF(8)=0
  OUTBF(9)=LIDFD(INBF(6))
  CALL LPOUT(9)
  IF(N.NE. 2) GOTO 999
  ACKSEQ(II)=INBF(2)
  999 RETURN
      END

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:32:22      PAGE 001

0001      SUBROUTINE LPINFT(LI)
0002      INTEGER*2 STAT,FLWCNT,OUTFCT,OUTO,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONITOR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRITM
0006      COMMON/DFM/ OUTBF(256),INBF(256),IWRITM,IWRT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTO,
1          RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0008      CC1=INBF(3)
0009      CC2=INBF(4)
0010      IF(CC1.LT. 0) GOTO 25
0011      IF(CC1.EQ. 0.AND. CC2.EQ. 0) GOTO 40
0012      CC3=MOD(CC1,2)
0013      CC4=MOD(CC1,256)
0014      IF ((CC3.GE. 1.OR. CC4.GE. 128).AND. ISENT.EQ. 1) GOTO 20
0015      IF ((INBF(1).EQ. 85).AND. INBF(2).EQ. 170) GOTO 99
0016      CC5=MOD(INBF(3),64)
0017      IF(CC5.GE. 32) CALL INPTQ(LI)
0018      GOTO 99
0019      20      CC7=MOD(CC1,2)
0020      IF(CC7.GE. 1) GOTO 30
0021      25      CALL ACKNAK(0)
0022      GOTO 99
0023      30      CALL ACKNAK(1)
0024      GOTO 99
0025      40      CALL INPTQ(LI)
0026      99      CONTINUE
0027      RETURN
0028      END
0029
0030
0031
0032
0033
0034

```



```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:32:37      PAGE 001

0001      SUBROUTINE INIT
0002      INTEGER*2 XING,XOUTQ,ACKQ,PXING,PXOUTQ,PACKQ,PINQ,FREE
0003      INTEGER*2 STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP,OUTQ,RESNLM,RSTBKP
0005      LOGICAL*1 DUM,IBFO,IBF1
0006      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0007      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID
0008      INTEGER*2 LTIME,OLDTIM,TIMLIM,ACKTIM,ATIMLM,IWRTIM
0009      COMMON/UFM/OUTBF(256),INBF(256),IWRTIM,IWRT,STAT,LTIME
0010      COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
1          ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0011      COMMON /MESS/ MESSEQ
0012      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
1          IOFLG,LLFLG,IRSEND
0013      COMMON /FRE/ FREE(64),IFR,IFRSZ
0014      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0015      COMMON      PACK(256,64)
0016      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0017      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0018      CALL RAM(0,7,4)
0019      CALL RAM(0,255,0)
0020      IFRSZ=64
0021      MESSEQ=0
0022      LLFLG=0
0023      INFCT=0
0024      ISLID=25
0025      IOFLG=0
0026      IRSEND=0
0027      CR=.015
0028      LF=.012
0029      ETX=.003
0030      RESNLM=2
0031      IQLNTH=16
0032      IALTRT=0
0033      FLWCNT=0
0034      IWRTIM=0
0035      TIMLIM=50
0036      SETPRM=4
0037      RSTPRM=32
0038      SETBKP=8
0039      RSTBKP=64
0040      ATIMLM=200
0041      MONTOR=24
0042      IWRT=0
0043      OUTFCT=0
0044      IFULL=0
0045      ISENT=0
0046      DO 40 I=1,20
0047      LIDFD(I)=4
0048      LIDFD(21)=1
0049      LIDFD(22)=3
0050      LIDFD(23)=6
0051      LIDFD(24)=5

```

PAGE 002

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:32:37

```

0052      LIDFD(25)=7
0053      LIDFD(26)=8
0054      LIDFD(27)=9
0055      LIDFD(28)=2
0056      DO 50 I=29,39
0057      LIDFD(I)=0
0058      DO 60 I=40,44
0059      LIDFD(I)=1
0060      DO 70 I=45,59
0061      LIDFD(I)=0
0062      DO 80 I=60,64
0063      LIDFD(I)=2
0064      DO 90 I=65,79
0065      LIDFD(I)=0
0066      DO 100 I=80,84
0067      LIDFD(I)=4
0068      DO 110 I=85,99
0069      LIDFD(I)=0
0070      DO 120 I=100,104
0071      LIDFD(I)=5
0072      DO 130 I=105,256
0073      LIDFD(I)=0
0074      PING(1)=IQLNTH
0075      PING(2)=IQLNTH+1
0076      PACKQ(1)=IQLNTH
0077      PACKQ(2)=IQLNTH+1
0078      PXOUTQ(1)=IQLNTH
0079      PXOUTQ(2)=IQLNTH+1
0080      PXING(1)=IQLNTH
0081      PXING(2)=IQLNTH+1
0082      DO 10 I=1,IFRSZ
0083      PACK(255,I)=0
0084      DO 20 I=1,IFRSZ
0085      FREE(I)=1
0086      DO 30 I=1,256
0087      ALKSEQ(I)=256
0088      DO 140 I=1,20
0089      ACKSEQ(I)=254
0090      IFR=IFRSZ
0091      RETURN
0092      END
0093

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 13:32:59      PAGE 001

0001      SUBROUTINE LINLOS
0002      REAL*8 RM1(5),LIN08,LIN18
0003      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM,PACKQ,ACKQ,I,STAT
0004      LOGICAL*1 OUTBF,INBF
0005      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0006      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0007      LOGICAL*1 M1(40),LINO(8),LIN1(8)
0008      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1 RESNLM,IALTRY,ISLID,MONITOR,LF,ETX,CR,DUM
0010      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0011      1 LIDFD(256),IOFLG,LLFLG,IRSEND
0012      COMMON /MESS/ MESSEQ
0013      DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0014      DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0015      DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0016      DATA RM1(5)/'DE 25 '//
0017      EQUIVALENCE(LINO,LIN08)
0018      EQUIVALENCE(LIN1,LIN18)
0019      IS=LLFLG
0020      DO 20 I=7,9
0021      XOUTBF(I)=LF
0022      CONTINUE
0023      DO 30 I=1,22
0024      XOUTBF(I+9)=M1(I)
0025      IF (IS.EQ.1) GOTO 50
0026      DO 40 I=1,8
0027      XOUTBF(I+31)=LINO(I)
0028      GOTO 70
0029      DO 60 I=1,8
0030      XOUTBF(I+31)=LIN1(I)
0031      DO 80 I=23,40
0032      XOUTBF(I+17)=M1(I)
0033      XOUTBF(58)=CR
0034      XOUTBF(59)=LF
0035      XOUTBF(60)=ETX
0036      CALL COUT(60)
0037      LLFLG=0
0038      RETURN
0039      END

```

PAGE 001

FORTRAN IV V02.1-1 Tue 05-Jun-79 13:31:21

```

0001 SUBROUTINE HINPUT
0002 INTEGER*2 XINQ,XOUTQ,ACKQ,FXINQ,FXOUTQ,PACKQ,PINQ
0003 INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNLM
0004 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID
0005 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,PACK,DUM
0006 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0007 LIDFD(256),IOFLG,LLFLG,IRSEND
0008 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009 RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0010 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0011 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0012 COMMON /MESS/ MESSEQ
0013 COMMON PACK(256,64)
0014 IF ((XINBF(1).EQ.'A').AND.
0015 (XINBF(2).EQ.'T').AND.
0016 (XINBF(3).EQ.'T').AND.
0017 (XINBF(4).EQ.'A').AND.
0018 (XINBF(5).EQ.'C').AND.
0019 (XINBF(6).EQ.'H')) GOTO 500
0020 CALL ENABLE(0)
0021 JJ=IGETSP(N)
0022 CALL ENABLE(1)
0023 DO 10 I=1,IOFLG
0024 PACK(I+6,JJ)=XINBF(I)
0025 10 CONTINUE
0026 IF(MESSEQ.EQ.126) MESSEQ=0
0027 MESSEQ=MESSEQ+1
0028 PACK(1,JJ)=0
0029 PACK(2,JJ)=MESSEQ
0030 PACK(3,JJ)=0
0031 PACK(4,JJ)=0
0032 PACK(5,JJ)=MONTOR
0033 PACK(6,JJ)=ISLID
0034 PACK(10FLG+8,JJ)=ETX
0035 CALL ENSTR(PACK(254,JJ),10FLG+8)
0036 CALL ENABLE(0)
0037 CALL ENQUE(PXINQ,XINQ,JJ)
0038 CALL ENABLE(1)
0039 IOFLG=0
0040 RETURN
0041 500 DECODE(2,510,XINBF(7),ERR=900) ITERM
0042 510 FORMAT(I2)
0043 MONTOR=ITERM
0044 IOFLG=0
0045 900 RETURN
0046 END

```

FDM.MACRO MACRO V03.01 5-JUN-79 13:33:26 PAGE 1

```

1  .TITLE FDM.MACRO
2  .SRITL NODE25
3  .IDENT /V3.0/
4  .GLOBL LIDINT,LID,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WTOKEN,RAM,STATUS,STATBO
6  .GLOBL LPOUT,RSTART,RDPNT,DESTR,ENSTR,LPINFT
7  .NLIST CND
8  .PSECT
9
10
11      R0      =%0
12      R1      =%1
13      R2      =%2
14      R3      =%3
15      R4      =%4
16      R5      =%5
17      SP      =%6
18      PC      =%7
19
20
21
22      !INTERFACE ADDRESS
23      BAR= 172410
24      WCR= 172412
25      CSR= 172414
26      IOBUF= 172416
27      OPREG= 172416
28      HRCR= 177560
29      HRBUF= 177562
30      HXCSR= 177564
31      HXBUF= 177566
32
33      !INTERFACE OPTIONS
34
35      XCIO=0
36      XSIO=1
37      XGIO=1
38      NDXX=0
39      ND24=1
40      NHOST=0
41      DHOST=1
42
43      !LOCAL VARIABLES
44
45      ARG1= 2
46      ARG2= 4
47      ARG3= 6
48      ARG4= 10
49
50      BIT15= 100000
51      BIT14= 40000
52      BIT08= 400
53      BIT07= 200
54      BIT06= 100
55      BIT05= 40
56      BIT04= 20
57      BIT03= 10
58      BIT02= 4

```


MACRO V03.01 5-JUN-79 13:33:26 PAGE 1-1

FDM.MACRO
NODE25

66
67
68

000002
000001

BIT01= 2
BIT00= 1

[illegible]


```
58 000730 132767 000002 000254 B1$: R1R: CAUSE  
59 000736 001423 RTI$  
60 000740 012701 000400' INBF, R1  
61 000744 012702 010610 $4488., R2  
62 000750 012703 010640 $4512., R3  
63 000754 012704 000002 $BIT01, R4  
64 000760 004767 000046 PC, ENBF  
65 000764 012705 001214' $AREA, R5  
66 000770 012767 000001 000216 $1, AREA  
67 000776 010267 000216 R2, DATA  
68 001002 004767 0000006 PC, LFINFT  
69  
70 001006 012605 (SP)+, R5  
71 001010 012604 (SP)+, R4  
72 001012 012603 (SP)+, R3  
73 001014 012602 (SP)+, R2  
74 001016 012601 (SP)+, R1  
75 001020 012600 (SP)+, R0  
76 001022 052737 040000 172414 $BIT14, $CSR  
77 001030 000002 RTI  
78  
79 001032 012737 010600 172416 EMEF: MOV $4480., $DFREG  
80 001040 012737 002400 172416 $1280., $DFREG  
81 001046 013700 172416 $IDBUF, R0  
82 001052 130400 BITB R4, R0  
83 001054 001002 CRCK  
84 001056 012704 177777 MOV $-1, R4  
85 001062 010237 172416 MOV R2, $DFREG  
86 001066 012737 001400 172416 $768., $DFREG  
87 001074 105737 172414 $CSR  
88 001100 100375 BFL -4  
89 001102 013702 172416 $IDBUF, R2  
90 001106 042702 177400 MOV $177400, R2  
91 001112 010200 MOV R2, R0  
92 001114 005400 NEG R0  
93 001116 010037 172412 MOV R0, $WCR  
94 001122 010137 172410 MOV R1, $BAR  
95 001126 010337 172416 MOV R3, $DFREG  
96 001132 012737 001400 172416 $768., $DFREG  
97 001140 105737 172414 $CSR  
98 001144 100375 BFL -4  
99 001146 012737 021000 172416 $8704., $DFREG  
100 001154 000240 NOP  
101 001156 105737 172414 $CSR  
102 001162 100401 DMACK  
103 001164 000240 NOP  
104 001166 012737 004400 172416 DMAOK: MOV $2304., $DFREG  
105 001174 105737 172414 $CSR  
106 001200 100375 BFL -4  
107 001202 005704 R4  
108 001204 100001 TST END$  
109 001206 010402 R2  
110 001210 000207 MOV R4, PC  
111 001212 000000 RTS 0  
112 001214 000000 CAUSE: .WORD 0  
113 001216 001220' AREA: .WORD 0  
114 001220 000000 DATA: .WORD 0
```

```
INBUF1 FULL  
NO MORE DONE  
BUFFER ADDRESS  
RDUFADR COMM  
SEL INBUF1  
CRC.BIT  
GO EMPTY BUFFER  
DATA LINK AREA  
ONE VARIABLE  
CRC OR BYTE COUNT  
CALL FORTRAN QUE'ER  
RESTORE REGISTERS
```

```
ENABLE INTERRUPTS  
RETURN FROM INTERRUPT
```

```
READ STATUS 1  
RS  
FETCH STATUS  
GOOD CRC
```

```
NO FLAG  
RDBUFADR  
RD  
GOOD RD
```

```
REPLACE WITH POINTER  
CLEAR MST BITS  
SAVE IT  
2'S COMP
```

```
BYTE COUNT NOW  
ADDRESS IN MEMORY  
SEL BUFFER  
FALSE RD  
DONE ON  
NO LOOP  
FIRE DMA  
DELAY  
GOOD DMA
```

```
ERROR IF HERE  
FALSE WD  
GOOD WD
```

```
WAS CRC OK  
YES  
NO FLAG IT  
RETURN
```

```
STATUS BYTE 0 HOLDER
```



```

1  .ENABLE LSB
2  *****
3  *****
4  *****
5  *****
6  *****
7  *****
8  *****
9  *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****

;WHICH SUBROUT.
;DISABLE LIU INTERRUPTS
;ENABLE LIU

;ENCODE BYTE COUNT
;DECODE BYTE COUNT

;FORTRAN ENCODE/DECODE MACRO
;CALL SWITCH(X) -SET OR RESET LINE SWITCHS
;
;04=SET PRIMARY LINE
;32=RESET PRIMARY LINE
;08=SET BACKUP LINE
;64=RESET BACKUP LINE

SWITCH: MOV @ARG1(R5),R1
MOV #4368., @#OFREG
ADD #2304., R1
MOV R1, @#OFREG
TSTB @#CSR
BPL -4
CLRB @#CSR
RTS PC

;SW SET
;WCR : MODSTAT
;RD
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE.BIT

```

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****

;*****
;***** PROCEDURE STATUS
;*****

;CALL STATUS(X,DATA) -READ STATUS BYTES 0/1
;
;X=0 STATUS BYTE 0
;X=1 STATUS BYTE 1

STATUS: MOV @ARG1(R5), R1
CMP #1, R1
BEQ 2$
MOV #4352., @#OPREG
MOV #1280., @#OPREG
MOV @#IOBUF, R0
BIC #177400, R0
MOV R0, @ARG2(R5)
RTS PC
MOV #4480., @#OPREG
MOV #1280., @#OPREG
MOV @#IOBUF, R0
BIC #177400, R0
MOV R0, @ARG2(R5)
RTS PC
MOV #4352., @#OPREG
MOV #1280., @#OPREG
MOV @#IOBUF, R1
BIC #177400, R1
BIT #BIT03, R1
BEQ 3$
MOV #1, @ARG1(R5)
RTS PC
MOV #0, @ARG1(R5)
RTS PC

;WHICH BYTE
;COMPARE
;WCR : RS(0)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;WCR : RS(1)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;STATUS BYTE 1
;RS
;FETCH DATA
;BUFFER FULL
;LOAD RETURN
;LOAD RETURN

```

AD-A078 391 BURROUGHS CORP PAOLI PA FEDERAL AND SPECIAL SYSTEMS GROUP F/G 9/2
SOFTWARE MAINTENANCE MANUAL FOR THE MODULAR SYSTEM CONTROL DEVE--ETC(U)
NOV 79 DCA100-76-C-0083
UNCLASSIFIED 66157 SBIE-AD-E100 313 NL

3 OF 5
AD
A078391



```

*****
**** PROCEDURE WRITE TOKEN (WTKEN) *****
*****

;CALL WTKEN -LOADS AND WRITES A TOKEN TO LOOP

;READ STATUS(1)
;RS
;FECH DATA
;BUFFER LOADED
;NO GENERATE TOKEN
;MODSTAT
;MARK OBOF/OBIF/WTCMD
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT

;SELECT BUFFER 1
;ADDRESS OF WRITE TOKEN
;# OF BYTES
;FORMAT IT
;DMA BYTE COUNT
;FIRE DMA
;DELAY
;GOOD DMA
;ERROR IF HERE
;CLEAR DONE
;MODSTAT
;MARK OBIF/WTCMD
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;RETURN
;A WTKEN

*****
**** PROCEDURE TIME (TIME) *****
*****

TIME: INC LTIME
CMP #077777, LTIME
BNE 4$
CLR LTIME
RTI

4$:

*****

```

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****

*****
***** PROCEDURE ACRAM *****
*****
;CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;0100 =DREAD
;0111 =NULL
;0000 =WTOKEN

RAM:
MOV @ARG1(R5), R0
CMP R0, R0
BNE RAM
WTRAM:
MOV @ARG2(R5), R0
MOV @ARG3(R5), R1
MOV @4354., @OPREG
ADD @2304., R0
MOV R0, @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV @4353., @OPREG
ADD @2304., R1
MOV R1, @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
RTS
PC
RDRAM:
MOV @ARG2(R5), R0
MOV @4354., @OPREG
ADD @2304., R0
MOV R0, @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV @4353., @OPREG
MOV @768., @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV @4353., @OPREG
MOV @768., @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV @IOBUF, R0
BIC @17760, R0
MOV R0, @ARG3(R5)
RTS
PC

; RAM
; WHICH OPERATION
; READ OP
; ADDRESS
; WRITE DATA
; SEL LDATA
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; WD/DATA (CMD)
; WRITE
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; ADDRESS TO READ
; SEL LDATA
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; READ DATA
; VALID READ
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; FETCH DATA
; CLEAR BITS
; RTN DATA

```


FDM.MACRO MACRO V03.01 5-JUN-79 13:33:26 PAGE 8

NODE25

```

1      .ENABLE LSR
2      *****
3      ***** PROCEDURE WRITE LOOP (LFOUT) *****
4      *****
5
6      LFOUT:  MOV     @ARG1(R5), R2
7              MOV     $340, R0
8              MTPS
9              MOV     @OUTBF, @BAR
10             NEG     R2
11             MOV     R2, @WCR
12             MOV     $4448, @OPREG
13             MOV     $10240, @OPREG
14             NOP
15             TSTB
16             NOP
17             CLRB
18             MOV     $4576, @OPREG
19             MOV     @WTD, @BAR
20             MOV     $2, R2
21             NEG     R2
22             MOV     R2, @WCR
23             MOV     $10240, @OPREG
24             NOP
25             TSTB
26             NOP
27             CLRB
28             MOV     $4368, @OPREG
29             MOV     $2307, @OPREG
30             TSTB
31             BPL     -4
32             CLRB
33             MOV     $000, R0
34             MTPS
35             RTS
36
37
38

```

; BYTE COUNT
 ; PRI=7
 ; BUS ADDRESS
 ; 2'S COMP COUNT
 ; COUNT
 ; OBO COMMAND
 ; DMA GO
 ; INTERFACE TIME
 ; DMA OK
 ; ERROR IF HERE
 ; OBI COMMAND
 ; ADDRESS OF WRITE TOKEN
 ; BYTE COUNT
 ; FORMAT FOR BLUI
 ; DMA BYTE COUNT
 ; FIRE DMA
 ; DELAY
 ; GOOD DMA
 ; ERROR IF NOT
 ; CLEAR DONE
 ; MODSTAT
 ; BUFFERS FULL
 ; GOOD WRITE
 ; NO LOOP UNTIL
 ; CLEAR DONE BIT
 ; PRI=0
 ; LOWER CPU

FDH-MACRO MACRO V03.01 5-JUN-79 13:33:26 PAGE 9
 NODE25

```

1 1 *****
2 2 *****
3 3 *****
4 4 *****
5 5 002274 017500 000002 000000 000000 000000 000000 000000 000000 000000
6 6 002300 022700 000000 000000 000000 000000 000000 000000 000000 000000
7 7 002304 001004 000000 000000 000000 000000 000000 000000 000000 000000
8 8 002306 012700 010410 000046 000001 000000 000000 000000 000000 000000
9 9 002312 000167 000046 000001 000000 000000 000000 000000 000000 000000
10 10 002316 022700 000001 000000 000000 000000 000000 000000 000000 000000
11 11 002322 001004 000000 000000 000000 000000 000000 000000 000000 000000
12 12 002324 012700 010610 000030 000002 000000 000000 000000 000000 000000
13 13 002330 000167 000030 000002 000000 000000 000000 000000 000000 000000
14 14 002334 022700 000002 000000 000000 000000 000000 000000 000000 000000
15 15 002340 001004 000000 000000 000000 000000 000000 000000 000000 000000
16 16 002342 012700 010510 000012 000004 000000 000000 000000 000000 000000
17 17 002346 000167 000012 000004 000000 000000 000000 000000 000000 000000
18 18 002352 022700 000004 000000 000000 000000 000000 000000 000000 000000
19 19 002356 001020 000000 000000 000000 000000 000000 000000 000000 000000
20 20 002360 012700 010710 000037 000004 000000 000000 000000 000000 000000
21 21 002364 010037 172416 000004 000000 000000 000000 000000 000000 000000
22 22 002370 012737 001400 000004 000000 000000 000000 000000 000000 000000
23 23 002376 105737 172414 000004 000000 000000 000000 000000 000000 000000
24 24 002402 100375 000004 000000 000000 000000 000000 000000 000000 000000
25 25 002404 013700 172416 000004 000000 000000 000000 000000 000000 000000
26 26 002410 042700 177400 000004 000000 000000 000000 000000 000000 000000
27 27 002414 010075 000004 000000 000000 000000 000000 000000 000000 000000
28 28 002420 000207 000004 000000 000000 000000 000000 000000 000000 000000

RDENT:  MOV @ARG1(R5), R0
        CMP #0, R0
        BNE 1$
        MOV #4360., R0
        JHP 4$
        CMP #1, R0
        BNE 2$
        MOV #4488., R0
        JHP 4$
        CMP #2, R0
        BNE 3$
        MOV #4424., R0
        JHP 4$
        CMP #4, R0
        BNE 5$
        MOV #4552., R0
        MOV R0, @#0PREG
        MOV #768., @#0PREG
        TSTB @#CSR
        BPL *-4
        MOV @#IOBUF, R0
        BIC #177400, R0
        MOV R0, @ARG2(R5)
        RTS PC

1$:
2$:
3$:
4$:
5$:

;FETCH COMMAND
;INBUFO ?
;NO
;RDBUFADR IN0
;INBUFO ?
;NO
;RDBUFADR IN1
;OUTBUFO ?
;NO
;RDBUFADR OUT0
;OUTBUFO ?
;NO RETURN
;RDBUFADR OUT1
;RDBUFADR
;RD
;READY
;NO LOOP UNTIL
;FETCH POINTER
;CLEAR MST
;RETURN POINTER

```

FUN.MACRO
NODE25

MACRO V03.01 5-JUN-79 13:33:26 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 002422 017500 000002 RSTART: MOV @ARG1(R5), R0
10 002426 022700 000000 RS0:  #0, R0
11 002432 001002 BNE RS1
12 002434 000000 HALT
13 002436 000207 RTS PC
14 002440 022700 000001 RS1:  #1, R0
15 002444 001003 BNE RS2
16 002446 013700 000040 G4:  @#40, R0
17 002452 000110 JMP (R0)
18 002454 022700 000002 RS2:  #2, R0
19 002460 001003 BNE RST
20 002462 013700 002472' MOV @#ROM, R0
21 002466 000110 JMP (R0)
22 002470 000207 RST: RTS PC
23 002472 173000 ROM: .WORD 173000
24
25
26

```

;MODE
 ;LOAD ADDRESS

 ;RESTART PROGRAM
 ;START ADDRESS
 ;LOAD MODE
 ;BOOT ADDRESS

```
1 1 .ENABLE LSR
2 2 .GLOBL HINFUT
3 3 *****
4 4 *****
5 5 *****
6 6 *****
7 7 CIO: MOV R0, -(SP) ;SAVE R0
8 8 MOV R1, -(SP) ;SAVE R1
9 9 MOV R2, -(SP) ;SAVE R2
10 10 MOV CPBUF, R0 ;BUFFER POINTER
11 11 MOV XINBF, R1 ;BUFFER ADDRESS
12 12 ADD R0, R1 ;CHAR OFFSET
13 13 TSTB @HRCR ;CHAR READY
14 14 BPL 7$
15 15 MOV @HRCR, R2
16 16 RIC $177600, R2 ;CLEAR PAR.
17 17 CMFB ;TURN ON CRT(REPORT)
18 18 BNE 20$
19 19 MOV R0, IXON ;IT'S ON
20 20 MOV @HXBUF, R0
21 21 NOP
22 22 NOP
23 23 MOV $007, R2
24 24 MOV @HXBUF, R2
25 25 BR 7$
26 26 CMFB R2 ;TURN OFF CRT(REPORT)
27 27 BNE 20$
28 28 MOV $2$, R2 ;IT'S OFF
29 29 MOV @HXBUF, R2
30 30 BR 7$
31 31 CMFB R2 ;DEL CHAR
32 32 BNE 2$
33 33 DEC R0 ;POINTER-1
34 34 MOV $134, R0 ;CHAR
35 35 JSR PC, ECHO
36 36 BR 7$
37 37 CMFB R2 ;STORE CHAR
38 38 BNE 4$
39 39 MOV R2, (R1)+ ;POINTER+1
40 40 INC R0 ;PRINT CHAR
41 41 JSR PC, CRLF ;LF CHAR
42 42 MOV $012, R0
43 43 INC R0 ;ATT PACKET READY
44 44 MOV $007, R0 ;RING BELL
45 45 JSR PC, IOFLG ;BYTE COUNT
46 46 MOV R0, R0
47 47 CLR R0
48 48 BR 7$
49 49 CMFB R2 ;ERASE LINE CHAR
50 50 BNE 5$
51 51 MOV CLP, R0
52 52 MOV $136, R0 ;RESTORE LINE POINTER
53 53 JSR PC, ECHO ;PRINT ~
54 54 MOV $125, R0 ;PRINT U
55 55 JSR PC, ECHO
56 56 JSR PC, CRLF ;PRINT CR LF
57 57 JSR PC, CRLF
```


FDM-MACRO MACRO V03.01 5-JUN-79 13:33:26 PAGE 11-1
NODE25

```

58 002756 000415 BR 7$
59 002760 122702 CMFB 5$
60 002764 001020 BNE 9$
61 002766 012721 MOV (R1)+
62 002772 005200 INC R0
63 002774 012721 MOV (R1)+
64 003000 005200 INC R0
65 003002 004767 JSR PC, CRLF
66 003006 010067 MOV R0, CLP
67 003012 010067 MOV R0, CPBUEF
68 003016 012602 MOV (SP)+, R2
69 003020 012601 MOV (SP)+, R1
70 003022 012600 MOV (SP)+, R0
71 003024 000002 RTI
72 003026 110221 MOV R2, (R1)+
73 003030 010267 MOV R2, CHAR
74 003034 005200 INC R0
75 003036 004767 JSR PC, ECHO
76 003042 000763 BR 7$
77 003044 105737 TSTB @HXCSR
78 003050 100375 ECHO
79 003052 116737 MOV CHAR, @HXBUF
80 003060 000207 RTS PC
81 003062 105737 TSTB @HXCSR
82 003066 100375 BPL CRLF
83 003070 012737 MOV (R1)+, @HXBUF
84 003076 105737 TSTB @HXCSR
85 003102 100375 BPL 10$
86 003104 012737 MOV (R1)+, @HXBUF
87 003112 000207 RTS PC
88
89
90
91 003114 017500 COUNT: MOV @ARG1(R5), R0
92 003120 012701 MOV @XOUTBF, R1
93 003124 062700 ADD #6, R0
94 003130 062701 ADD #6, R1
95 003134 105737 TSTB @HXCSR
96 003140 100375 BPL 11$
97 003142 112137 MOV (R1)+, @HXBUF
98 003146 122711 CMPB @003, (R1)
99 003152 001401 REG 12$
100 003154 077011 SOB R0, 11$
101 003156 000207 RTS PC
102 003160 000000 CLP: .WORD 0
103 003162 000000 CPBUEF: .WORD 0
104 003164 000000 CHAR: .WORD 0
105 003166 000000 DUMP: .BLKW 50.
226

```

;EOP CHAR
 ;CR
 ;LF
 ;FLAG A NEW LINE
 ;RESTORE R2
 ;RESTORE R1
 ;RESTORE R0
 ;BYTE COUNT
 ;BUFFER
 ;OFFSET PACKET
 ;WRITE CHAR
 ;END YET
 ;YES STOP PRINTING

FDM,MACRO
NODE25

MACRO V03.01 5-JUN-79 13:33:26 PAGE 12

```

1 *****
2 *****
3 *****
4 *****
5 000000 .PSECT IFM,RW,D,GBL,REL,OVR
6
7 000000 OUTBF: .BLKB 256.
8 000400 INBF: .BLKB 256.
9 001000 IWRTTH: .BLKW
10 001002 IWRT: .BLKW
11 001004 STAT: .BLKW
12 001006 LTIME: .BLKW
13 .EVEN
14
15 000000 .PSECT BUFS,RW,D,GBL,REL,OVR
16
17 000000 XOUTBF: .BLKB 256.
18 000400 XINBF: .BLKB 256.
19 001000 ACKSED: .BLKB 256.
20 001400 LIDFD: .BLKB 256.
21 002000 IOFLG: .BLKW
22 002002 LLFLG: .BLKW
23 002004 IRSEND: .BLKW
24
25
26 000000 .PSECT CRT,RW,D,GBL,REL,OVR
27
28 000000 IXON: .BLKW
29 .EVEN
30
31
32 000001 .END

```

FDM-MACRO MACRO V03.01 5-JUN-79 13:33:26 PAGE 12-1

SYMBOL TABLE

ACKREQ	001000R	003 B2\$	000516R	ECHO	003044R	LIDFD	001400R	003 RS1	002440R
AREA	001214R	B3\$	000542R	EMBF	001032R	LIO	000436RG	RS2	002454R
ARG1	= 000002	B4\$	000566R	ENABLE	001222RG	LIUINT	000104RG	RT1\$	001006R
ARG2	= 000004	B5\$	000614R	END\$	001210R	LLFLG	002002R	003 STAT	001004R
ARG3	= 000006	B6\$	000632R	ENSTR	001254RG	LFINPT=	***** G	STATB0	001430RG
ARG4	= 000010	B7\$	000576R	E0\$	001234R	LFOUT	002110RG	STATUS	001332RG
BAR	= 172410	CAUSE	001212R	E1\$	001244R	LTIME	001006R	002 STAT\$	000412R
BIT00	= 000001	CHAR	003164R	G4\$	002446R	MASTER	000000RG	SWITCH	001274RG
BIT01	= 000002	CIO	002474RG	HINPOT=	***** G	NDXX	= 000000	TIME	001662RG
BIT02	= 000004	CLP	003160R	HRBUF	= 177562	ND24	= 000001	WCR	= 172412
BIT03	= 000010	COUT	003114RG	HRCR	= 177560	NHOST	= 000000	WTD	001660R
BIT04	= 000020	CPBUF	003162R	HXBUF	= 177566	DFREG	= 172416	WTKG	001560R
BIT05	= 000040	CRCOK	001062R	HXCSR	= 177564	OUTBF	000000R	002 WTKEN	001502RG
BIT06	= 000100	CRLF	003062R	INBF	000400R	002 RAM	001704RG	WTRAM	001716R
BIT07	= 000200	CSR	= 172414	IOBUF	= 172416	KDPNT	002274RG	XCIO	= 000000
BIT08	= 000400	DATA	001220R	IOFLG	002000R	003 KDRAM	002010R	XGIO	= 000001
BIT14	= 040000	DESTR	001264RG	IRSEND	002004R	003 ROM	002472R	XINBF	000400R
BIT15	= 100000	DHOST	= 000001	IWRT	001002R	002 RST	002470R	XOUTBF	000000R
B0\$	000652R	DMAOK	001166R	IWRTM	001000R	002 RSTART	002422RG	XSIO	= 000001
B1\$	000730R	DUMP	003166R	IXON	000000R	004 RS0	002426R	ZEROBP	000312R

• ABS. 000000 000
 00332 001
 DFM 001010 002
 BUFS 002006 003
 CRT 000002 004
 ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 306 WORDS (2 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 50 PAGES
 DK:FDM,DK:FDM=DK:FDM

1603 061

12:00:00

01 00 79

1000 00.000

1000 00.000

1000 00.000

FOR 0000000000 100
FOR 0000000000 100
000 100

PAGE 001

12:00:00

01 JUN 79

100.25.000

PROGRAM:

ACQUAT-1

ASSIGN DCO: DL:

K L IN

DL:RUB 25, DL:RUB=DL:RUB / 1/1.4

DL:FB/C

DL:RUB 1/1

45HRT

ASSIGN DL: DL:

1.7 NODE 26 (BWBSA)

The Base Band Signal Analysis and Wide Band Signal Analysis (BWBSA) module assesses the performance of link equipment. Three links are monitored. There are seven parameters to be checked per link for the baseband, and six for the wideband:

1. Transmitter Percent Modulation (BTPM)
2. Transmitter Frequency Deviation (BTFD)
3. Relative Transmitter Power (B RTP)
4. Receiver AGC Voltage (BRAV)
5. Receiver IF Output (BRIO)
6. Multiplex Baseband Levels (BMBL)
7. Multiplex Pilot Levels (BMPL)
8. Transmitter Percent Deviation (WTFD)
9. Transmitter Frequency Deviation (WTFD)
10. Relative Transmitter Power (WRTP)
11. Receiver AGC Voltage (WRAV)
12. Receiver IF Output (WRIO)
13. Multiplexor Pseudo Error Rate (WMPE)

The SIG produces simulated values for three parameters. In addition, alarms are generated associated with the three links for transmitters, receivers, and multiplexers. Alarms and Red and Amber threshold values are sent to the FIAC as Event Reports. Event Reports consist of one-byte link number, 2-byte baseband number, 2-byte wideband number, one-byte supergroup, one-byte group, one-byte condition, a 3-byte monitor point number, and the node designator to which reports should be sent.

The BWBSA interprets commands from the DBMS similar to the VSQC and DSQC except that measurements are performed for 3 links rather than channels.

1.7.1 Program Descriptions

1.7.1.1 Refer to Section 1.1 for description of routines NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME, RAM, LPOUT, RDPNT, RSTART.

1.7.1.2 Subroutine BWBSA (FORTRAN)

This subroutine is called from the nodal program when a message has been received from the SIG. The inputs are compared with the thresholds and subroutine SNDRPT is called if a measurement is outside the threshold.

1.7.1.3 Subroutine SNDRPT (FORTRAN)

This subroutine is called by routine BWBSA to send the FIAC node a message identifying the equipment that failed and the value of measurement.

1.7.1.4 Subroutine BWSALP (FORTRAN)

This subroutine is called by nodal when a message is received from the loop. It performs three functions: reporting on or off, and take a measurement of a link.

1.7.1.5 Subroutine SIO (MACRO)

This subroutine reads the simulated inputs from the SIG. It is called by an interrupt and when the message is read, it sets a flag containing the byte count.

1.7.1.6 Subroutine SOUT (MACRO)

This subroutine is used to send the SIG a link number for a requested measurement.

1.7.1.7 Subroutine READY (MACRO)

This subroutine is used to signal the SIG that the BWBSA module is ready for another measurement.

```

PORTRAN IV      V02.1-11      Mon 01-Oct-79 09:20:30      PAGE 001

0001      PROGRAM NOTAL
0002      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ
0003      INTEGER*2 PINQ,PRE,STAT,FLWNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNM,OUTQ,DEQUE
0006      LOGICAL*1 YOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0007      LOGICAL*1 IDATA
0008      LOGICAL*1 PACK,RTX,CR,LF,MONITOR,ISLID,IUM
0009      INTEGER*2 ITIME,ACTIM,ME*TIM,OLDTIM,IWRTM,TIMLIM,ATIMLM
0010      REAL*4 RH,RL,AH,AL,VMEAS
0011      COMMON /MESS/ MESSIG
0012      COMMON /DFM/ OUTBF(256),INBF(256),
0013      1      IWRTM,IWRT,STAT,ITIME
0014      1      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0015      1      COMMON /BUPS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0016      1      LIDFD(256),IOFLG,LLFLG,IRSEND
0017      COMMON /PRE/ PRE(64),IPR,IPRSZ
0018      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0019      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0020      COMMON /GLOB/ ISENT,FLWNT,ISLID,OUTFCT,IFULL,OUTQ,
0021      1      RESNM,IALIRT,ISLID,MONITOR,LF,RTX,CR,DUM
0022      5      COMMON /BCOM/ ISWTR,RH(13,3),RL(13,3),AH(13,3),AL(13,3),
0023      5      ITERM,LINK,MON,VMEAS(13),ISUB,INUM,IDATA(10),
0024      5      ICHNSW
0025      CALL MASTER
0026      CALL INIT
0027      CALL ENABLE(1)
0028      5      CALL READY
0029      25      IF(PINQ(1) .LT. PINQ(2)) GOTO 40
0030      CALL ENABLE(3)
0031      Q2=DEQUE(PINQ,INQ,1)
0032      CALL ENABLE(1)
0033      DO 30 I=1,10
0034      30      IDATA(I)=PACK(I+6,Q2)
0035      CONTINUE
0036      CALL BWSALP
0037      IPR=IPR+1
0038      FREE(IPR)=Q2
0039      40      CALL STAT0(IIS)
0040      IF(IIS .EQ. 1) GOTO 100
0041      IF(IOFLG .GT. 2 .AND. ISEND .EQ. 0) CALL EMBSA
0042      IF(LLFLG .GT. 0) CALL LINLOS
0043      80      IF(OUTFCT .EQ. 1) GOTO 100
0044      IF((IFULL .EQ. 1) .OR. (ISENT .EQ. 1)) GOTO 120
0045      IF(IRSEND .EQ. 1) GOTO 85
0046      IF(PXINQ(1) .LT. PXINQ(2)) GOTO 100
0047      CALL ENABLE(2)
0048      Q1=DEQUE(PXINQ,XINQ,1)
0049      CALL ENABLE(1)
0050      OUTQ=Q1
0051      85      CALL DESTH(PACK(254,OUTQ),Q2)
0052      DO 90 I=1,Q2
0053      90
0054
0055
0056

```

PAGE 022

```

FORTRAN IV      V02.1-11   Mon 01-Oct-79 09:00:30
0057      OUTBF(1)=PACK(1,OUTQ)
0058      CONTINUE
0059      OUTBF(Q2+1)=0
0060      IPT=OUTBF(5)
0061      OUTBF(Q2+2)=LIDFD(IPT)
0062      IFULL=1
0063      ISENT=1
0064      CALL LPOUT(Q2+2)
0065      ISEND=1
0066      ACKTIM=0
0067      IWRTIM=0
0068      IWRT=1
0069      INPCT=0
0070      OLDTIM=NEWTIM
0071      NEWTIM=LTIME
0072      IF(IWRT.EQ.0) GOTO 120
0073      IWRTIM=IWRTIM+(NEWTIM-OLDTIM)
0074      IF(IWRTIM.LT. TIMLIN) GOTO 120
0075      CALL WTKEN
0076      IWRTIM=0
0077      IWRT=0
0078      IF(ISENT.EQ.0) GOTO 130
0079      ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0080      IF(ACKTIM.LT. ATIMLM) GOTO 130
0081      CALL ENABLE(0)
0082      CALL ACKNAK(0)
0083      CALL ENABLE(1)
0084      CONTINUE
0085      GOTO 5
0086      END
0087
0088      100 INPCT=0
0089      110 OLDTIM=NEWTIM
0090      NEWTIM=LTIME
0091      IF(IWRT.EQ.0) GOTO 120
0092      IWRTIM=IWRTIM+(NEWTIM-OLDTIM)
0093      IF(IWRTIM.LT. TIMLIN) GOTO 120
0094      CALL WTKEN
0095      IWRTIM=0
0096      IWRT=0
0097      IF(ISENT.EQ.0) GOTO 130
0098      ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0099      IF(ACKTIM.LT. ATIMLM) GOTO 130
0100      CALL ENABLE(0)
0101      CALL ACKNAK(0)
0102      CALL ENABLE(1)
0103      CONTINUE
0104      GOTO 5
0105      END

```

PAGE 001

V02.1-11 Mon 01-Oct-79 09:00:38

FORTRAN IV

```

0001 FUNCTION IGETSP(N)
0002 LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003 INTEGER*2 FREE,FLWNT
0004 INTEGER*2 OUTFCT,OUTQ,RESNLM
0005 COMMON /PRE/ PREB(64),IFR,IPPSZ
0006 COMMON /GLOB/ ISENT,FLWNT,IGLNT,OUTFCT,IFULL,OUTQ,
1 RESNLM,IALTBT,ISLID,MONITOR,LF,ETX,CR,DUM
0007 IF (IFR .LT. 1) CALL INIT
0008 IGETSP=FREE(IFR)
0009 IFR=IFR-1
0010 RETURN
0011 END
0012

```



```

FORTRAN IV      V02.1-11   Mon 01-Oct-79 09:00:41      PAGE 001

0001      SUBROUTINE ENQUE(A,B,N)
0002      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INO,PINQ
0004      INTEGER*2 PLACAT,OUTQ,A(2),E(16)
0005      INTEGER*2 RESNM,OUTFCT
0006      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007      1      ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0008      1      COMMON /GLOB/ ISENT,FLWNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0009      1      RESNM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0010      IQHEAD=A(1)
0011      IQTAIL=A(2)
0012      IF(IQTAIL.EQ. 1) GOTO 20
0013      IQTAIL=IQTAIL-1
0014      B(IQTAIL)=N
0015      A(2)=IQTAIL
0016      GOTO 999
0017      20 IF(IQHEAD.GE.(IQLNTH)) GOTO 40
0018      NN=IQHEAD-IQTAIL
0019      DO 30 I=1,NN+1
0020      30 B(IQLNTH+1-1)=B(IQHEAD+1-1)
0021      A(1)=IQLNTH
0022      A(2)=IQLNTH-NN
0023      GOTO 10
0024      40 CALL INIT
0025      999 RETURN
0026      END

```

```

FORTRAN IV      V02:1-11   Mon 01-Oct-79 09:00:44      PAGE 001

0001      FUNCTION DEQUE(A,B,ID)
0002      LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONITOR
0003      INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINC
0004      INTEGER*2 FLACNT,A(2),B(16),LETC,OUTFCT,OUTQ,RESNLM
0005      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      1 COMMON /GLOB/ ISENT,FLACNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0007      1 RESNLM,IALTET,ISLID,MONITOR,LF,ETX,CR,DUM
0008      IQHEAD=A(1)
0009      DEQUE=B(IQHEAD)
0010      IF(ID.NE.1) GOTO 999
0011      IF(IQHEAD.NE.0) GOTO 10
0012      A(1)=IQLNTH
0013      A(2)=IQLNTH+1
0014      GOTO 999
0015      10 A(1)=IQHEAD-1
0016      999 RETURN
0017      END
0018

```

```

X0FORTRAN IV      V02.1-11      Mon 01-Oct-79 09:00:51      PAGE 001

0001      SUBROUTINE ACKNAK(N)
0002      INTEGER*2 FLWCT,OUTFCT,STAT,XINC,PXINC,PXOUTQ
0003      INTEGER*2 ACKQ,PACKQ,PINC,FREE,OUTQ,XOUTQ,FINDPK
0004      INTEGER*2 T1,T2,T3,T4,T5,RESNLM
0005      LOGICAL*1 FTX,CR,LF,MONITOR,ISLID,LUM
0006      LOGICAL*1 PACK,OUTF,INBF,LCOMT(40)
0007      LOGICAL*1 XOUTF,XINBF,ACKSEQ,LIDFD
0008      INTEGER*2 LTIME,IWRTIM
0009      REAL*8 COMT(5)
0010      EQUIVALENCE(COMT,LCOMT)
0011      DATA COMT(1)/'MESSAGE',/,COMT(2)/'NOT SENT',/,COMT(3)/' FROM ',/,
0012      DATA COMT(4)/'NODE',/,COMT(5)/' TO NODE',/
0013      COMMON /GLOB/ ISENT,FLWCT,IQINTE,OUTFCT,IFULL,OUTQ,
0014      RESNLM,IALIRT,ISLID,MONITOR,LF,FTX,CR,DUM
0015      COMMON /DPM/ OUTBF(256),INBF(256),
0016      IWRTIM,IWRT,STAT,LTIME
0017      COMMON /QUE/ XINQ(16),PXINC(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0018      PACKQ(2),INQ(16),PINC(2)
0019      COMMON /FREE/ FREE(64),IFR,IFRSZ
0020      COMMON /MESS/ MESSEQ
0021      COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0022      IOPLG,LLPLG,IRSEND
0023      IF (N.NE.1) GOTO 10
0024      IFR=IFR+1
0025      DO 5 I=1,256
0026      PACK(I,OUTQ)=0
0027      CONTINUE
0028      IFULL=0
0029      ISENT=0
0030      IRSEND=0
0031      FREE(IFR)=OUTQ
0032      GOTO 999
0033      10 T2=PACK(256,OUTQ)
0034      T2=T2+1
0035      IF(T2.LE.RESNLM) GOTO 120
0036      T3=PACK(3,OUTQ)
0037      T4=MOD(T3,64)
0038      IF(T4.LT.32) GOTO 110
0039      T5=IGETSP(N)
0040      IF(MESSEQ.EQ.126) MESSEQ=0
0041      MESSEQ=MESSEQ+1
0042      PACK(1,T5)=0
0043      PACK(2,T5)=MESSEQ
0044      PACK(3,T5)=0
0045      PACK(4,T5)=0
0046      PACK(5,T5)=25
0047      PACK(6,T5)=ISLID
0048      DO 20 I=7,60
0049      PACK(I,T5)='040'
0050      20 CONTINUE
0051      DO 30 I=7,9
0052      PACK(I,T5)=LF
0053      30 CONTINUE

```

PAGE 002

FORTRAN IV V02.1-11 Mon 01-Oct-79 09:00:51

```

0054 DO 40 I=1,8
0055   PACK(I+9,T5)=LCOMT(I)
0056   CONTINUE
0057 DO 50 I=9,15
0058   PACK(I+9,T5)=LCOMT(I)
0059   CONTINUE
0060 DO 60 I=17,22
0061   PACK(I+9,T5)=LCOMT(I)
0062   CONTINUE
0063 DO 70 I=25,29
0064   PACK(I+7,T5)=LCOMT(I)
0065   CONTINUE
0066 ENCODE(3,00,PACK(30,T5)) ISLID
0067 80 FORMAT(I3)
0068 DO 90 I=33,40
0069   PACK(I+11,T5)=LCOMT(I)
0070   CONTINUE
0071 ENCODE(3,00,PACK(52,T5))PACK(5,OUTQ)
0072   PACK(55,T5)=LF
0073   PACK(56,T5)=LF
0074   PACK(57,T5)=CR
0075   PACK(58,T5)=ETX
0076   CALL ENSTR(PACK(254,T5),58)
0077   CALL ENQUE(PXINQ,XINQ,T5)
0078 DO 102 I=1,256
0079   PACK(I,OUTQ)=0
0080   CONTINUE
0081   ISENT=0
0082   IREND=0
0083   IFULL=0
0084   IWRT=0
0085   IFR=IFR+1
0086   FREE(IFR)=OUTQ
0087   GOTO 999
0088 110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089   PACK(256,OUTQ)=0
0090   ISENT=0
0091   IFULL=0
0092   IF(IREND.EQ.1) GOTO 999
0093   CALL ENQUE(PXINQ,XINQ,OUTQ)
0094   GOTO 999
0095 120 PACK(256,OUTQ)=T2
0096   IFULL=0
0097   ISENT=0
0098   IF(IREND.EQ.1) GOTO 999
0099   CALL ENQUE(PXINQ,XINQ,OUTQ)
0100   RETURN
0101 999
0102   END
0103

```

PAGE 001

V02.1-11 Mon 01-Oct-79 09:01:00

FORTRAN IV

```

0001 SUBROUTINE INPTQ(L)
0002 INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ
0003 INTEGER*2 PACKQ,PINQ,PLCNT,OUTFCT,T1,RESNLM,OUTQ
0004 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDF,OUTBF,INBF
0005 LOGICAL*1 PACK,MONTR,ISLID,ETX,CR,LF,DUM
0006 INTEGER*2 LTIME,IWRITM
0007 COMMON /DFM/ OUTBF(256),INBF(256),
0008          IWRITM,IWRT,STAT,LTIME
0009 1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010          LIDF(256),IOFLG,LLFLG,IRSEND
0011 1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012          ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0013 1 COMMON /GLOB/ ISENT,PLCNT,IQINTH,OUTFCT,IFULL,OUTQ,
0014          RESNLM,IALTRT,ISLID,MONTR,LF,ETX,CR,DUM
0015 IF(L.LT.1) GOTO 30
0016 LI=L-2
0017 II=INBF(6)
0018 IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
0019 T1=IGETSP(N)
0020 DO 10 I=1,LI
0021 PACK(I,T1)=INBF(I)
0022 CALL ENSTR(PACK(254,T1),LI)
0023 CALL ENQUE(PINQ,INQ,T1)
0024 N=2
0025 GOTO 40
0026 20 N=1
0027 GOTO 40
0028 30 N=0
0029 40 DO 50 I=1,2
0030          OUTBF(I)=INBF(I)
0031          CONTINUE
0032 NN=1
0033 IF(N.EQ.0) NN=128
0034 OUTBF(3)=NN
0035 OUTBF(4)=0
0036 OUTBF(5)=INBF(6)
0037 OUTBF(6)=ISLID
0038 OUTBF(7)=ETX
0039 OUTBF(9)=LIDF(INBF(6))
0040 CALL LPOUT(9)
0041 IF(N.NE.2) GOTO 999
0042 ACKSEQ(II)=INBF(2)
0043 999 RETURN
0044      END
0045

```



```

FORTRAN IV      V02.1-11   Mon 01-Oct-79 09:01:05      PAGE 001

0001      SUBROUTINE LPRINT(LI)
0002      INTEGER*2 STAT,FLWCT,OUTFCT,OUTQ,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUIBF,INEF,ETX,CR,LF,MONTR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRTTM
0006      COMMON/DFM/ OUTBF(256),INBF(256),IWRTTM,IWRT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,FLWCT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      1 RESNLM,IALTRT,ISLID,MONTR,IF,ETX,CR,DUM
0009      CC1=INEF(3)
0010      CC2=INEF(4)
0011      IF(CC1 -LT. 0) GOTO 25
0012      IF(CC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0013      CC3=MOD(CC1,2)
0014      CC4=MOD(CC1,256)
0015      IF ((CC3 .GE. 1 .OR. CC4 .GE. 128) .AND. ISENT .EQ. 1) GOTO 20
0016      IF ((INEF(1) .EQ. 85) .AND. INEF(2) .EQ. 170) GOTO 99
0017      CC5=MOD(INEF(3),64)
0018      IF(CC5 .GE. 32) CALL INPTQ(LI)
0019      GOTO 99
0020
0021      20      CC7=MOD(CC1,2)
0022      IF(CC7 .GE. 1) GOTO 30
0023      25      CALL ACKNAK(0)
0024      GOTO 99
0025      30      CALL ACKNAK(1)
0026      GOTO 99
0027      40      CALL INPTQ(LI)
0028      99      CONTINUE
0029      RETURN
0030      END
0031
0032
0033
0034

```

```

PORTRAN IV      V02.1-11      Mon 01-Oct-79 09:01:10      PAGE 001

0001      SUBROUTINE INIT
0002      REAL*4 RH,RL,AH,AL,VMEAS
0003      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ,FREE
0004      INTEGER*2 STAT,LLCNT,OUTFCT
0005      INTEGER*2 SETPRM,RSTPRM,SETBKF,OUTQ,RESNLM,RSTBKF
0006      LOGICAL*1 IDATA,DUM
0007      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDED,OUTBF,INBF
0008      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID
0009      INTEGER*2 LTIME,OLDTIM,TIMLIM,ACKTIM,ATIMLM,I*RTTM
0010      COMMON /DPM/ OUTBF(256),INBF(256),
0011      I*RTTM,I*RT,STAT,LTIME
0012      1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0013      1 COMMON /MESS/ ACKQ(16),PACKQ(2),INC(16),PINQ(2)
0014      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDED(256),
0015      1 IOFLG,LLFLG,IRSEND
0016      COMMON /PRE/ FREE(64),IFR,IPRSZ
0017      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0018      1 COMMON /SWT/ SETPRM,RSTPRM,SETBKF,RSTBKF
0019      1 COMMON /GLOB/ ISENT,FLWNT,ICLNTH,OUTFCT,IFULL,OUTQ,
0020      RESNLM,I*RT,ISLID,MONTOR,LF,ETX,CR,DUM
0021      1 COMMON /BCOM/ ISWTCR,RH(13,3),RL(13,3),AH(13,3),AL(13,3),
0022      1 ITERM,LINK,MON,VMEAS(13),ISUB,INUM,IDATA(12),
0023      1 ICHNSW
0024      CALL RAM(0.8,4)
0025      CALL RAM(0.255,0)
0026      MESSEQ=0
0027      IOFLG=0
0028      IRSEND=0
0029      LLFLG=0
0030      IFRSZ=64
0031      INFLCT=0
0032      ISLID=26
0033      CR="015
0034      LF="012
0035      ETX="003
0036      RESNLM=2
0037      ICLNTH=16
0038      I*RT=0
0039      FLWNT=0
0040      I*RTIM=0
0041      TIMLIM=30
0042      SETPRM=4
0043      RSTPRM=32
0044      SETBKF=8
0045      RSTBKF=64
0046      ATIMLM=200
0047      MONTOR=27
0048      I*RTI=0
0049      ISWTCR=0
0050      OUTFCT=0
0051      IFULL=0
0052      ISENT=0

```

```

FORTRAN IV      V02.1-11      Mon 21-Oct-79 09:01:12      PAGE 002

2049      DO 40 I=1,20
0050      LIDFD(I)=4
0051      LIDFD(21)=1
2052      LIDFD(22)=3
0053      LIDFD(23)=6
2054      LIDFD(24)=5
0055      LIDFD(25)=7
0056      LIDFD(26)=8
0057      LIDFD(27)=9
0058      LIDFD(28)=2
0059      DO 50 I=29,39
0060      LIDFD(I)=0
2061      DO 60 I=40,44
0062      LIDFD(I)=1
2063      DO 70 I=45,59
0064      LIDFD(I)=0
2065      DO 80 I=60,64
0066      LIDFD(I)=2
0067      DO 90 I=65,79
0068      LIDFD(I)=0
0069      DO 100 I=80,84
0070      LIDFD(I)=4
0071      DO 110 I=85,99
0072      LIDFD(I)=0
2073      DO 120 I=100,104
0074      LIDFD(I)=5
2075      DO 130 I=105,256
0076      LIDFD(I)=0
0077      PINQ(1)=IQLNTH
2078      PINQ(2)=IQLNTH+1
0079      PACKQ(1)=IQLNTH
2080      PACKQ(2)=IQLNTH+1
0081      PXOUTQ(1)=IQLNTH
2082      PXOUTQ(2)=IQLNTH+1
0083      PXINQ(1)=IQLNTH
2084      PXINQ(2)=IQLNTH+1
0085      DO 10 I=1,IPRSZ
0086      PACK(255,I)=0
2087      DO 20 I=1,IPRSZ
0088      FREE(I)=1
2089      DO 30 I=1,256
0090      ACKSEQ(I)=256
0091      IPR=IPRSZ
0092      RETURN
0093      END
0094

```

```

FORTRAN IV      V02.1-11      Mon 01-Oct-79 09:01:19      PAGE 001

0001      SUBROUTINE LINLOS
0002      REAL*8 RM1(5), LINQ0, LIN18
0003      INTEGER*2 ELWCNT, OUTFCT, OUTQ, RESNLM, T1
0004      INTEGER*2 XINQ, PXINQ, XOUTQ, ACKQ, PACKQ, PINQ
0005      LOGICAL*1 ETX, CR, LF, MONTOR, ISLID, DUM
0006      LOGICAL*1 PACK, M1(40), LINQ(8), LIN1(8)
0007      LOGICAL*1 XOUTEF, XINBF, ACKSEQ, LIDFD
0008      COMMON/QUE/XINQ(16), PXINQ(2), XOUTQ(16), PXOUTQ(2), ACKQ(16),
0009      &      PACKQ(2), INQ(16), PINQ(2)
0010      COMMON/BUFS/XOUTBF(256), XINBF(256), ACKSEQ(256), LIDFD(256),
0011      &      IOFLG, LFLG, IREND
0012      COMMON /GLOB/ ISENT, FLWCNT, IQLNTH, OUTFCT, IFULL, OUTQ,
0013      1      RESNLM, IALHT, ISLID, MONTOR, LF, ETX, CR, DUM
0014      COMMON /MESS/ MESSEQ
0015      DATA LINQ0, LIN18/'PRIMARY ','BACKUP '//
0016      DATA RM1(1), RM1(2)/'LOSS OF ','MODULATI'//
0017      DATA RM1(3), RM1(4)/'ON ON LO','OP AT NO'//
0018      DATA RM1(5)/'DE 26'//
0019      EQUIVALENCE(LINQ, LINQ0)
0020      EQUIVALENCE(LIN1, LIN18)
0021      IS=LLFLG
0022      CALL ENABLE(0)
0023      T1=IOETSP(N)
0024      CALL ENABLE(1)
0025      PACK(1, T1)=0
0026      IF(MESSEQ.EQ. 126) MESSEQ=0
0027      MESSEQ=MESSEQ+1
0028      PACK(2, T1)=MESSEQ
0029      PACK(3, T1)=0
0030      PACK(4, T1)=0
0031      PACK(5, T1)=25
0032      PACK(6, T1)=ISLID
0033      DO 20 I=7, 9
0034      PACK(I, T1)=LF
0035      CONTINUE
0036      DO 30 I=1, 22
0037      PACK(I+9, T1)=M1(I)
0038      IF(IS.EC. 1) GOTO 50
0039      DO 40 I=1, 8
0040      PACK(I+31, T1)=LINQ(I)
0041      GOTO 70
0042      DO 60 I=1, 8
0043      PACK(I+31, T1)=LIN1(I)
0044      DO 80 I=23, 40
0045      PACK(I+17, T1)=M1(I)
0046      PACK(58, T1)=CR
0047      PACK(59, T1)=LF
0048      PACK(60, T1)=ETX
0049      CALL ENSTR(PACK(254, T1), 60)
0050      CALL ENABLE(0)
0051      CALL ENQUE(PXINQ, XINQ, T1)
0052      CALL ENABLE(1)
0053

```

PAGE 002

V02.1-11 Mon 21-Oct-79 09:21:19

FORTRAN IV

0054
0055
0056
LLFC=0
RETURN
END

[illegible]

PAGE 002

FORTRAN IV V02.1-11

```

0030 DECODE(4,2,XINBF(43),ERR=80) VMEAS(11)
0031 DECODE(4,2,XINBF(47),ERR=80) VMEAS(12)
0032 DECODE(4,2,XINBF(51),ERR=80) VMEAS(13)
0033 DECODE(2,1,XINBF(55),ERR=90) LINE
0034 DECODE(2,1,XINBF(59),ERR=80) MON
0035 1 FORMAT(A2)
0036 2 FORMAT(A4)
0037 IF(ICHNSW.EQ.1) GOTO 72
0039 10 IF(ISUB.GT.3) ISUB=0
0040 ISUB=ISUB+1
0041 DO 20 INUM=1,13
0042 IF((AL(INUM,ISUB).LE.VMEAS(INUM)).AND.
0043 & (AL(INUM,ISUB).GE.VMEAS(INUM))) GOTO 20
0044 GOTO 40
0045 20 CONTINUE
0046 IOFLG=0
0047 RETURN
0048 40 IF((AL(INUM,ISUB).LT.VMEAS(INUM).AND.
0049 & RH(INUM,ISUB).GE.VMEAS(INUM)).OR.
0050 & (AL(INUM,ISUB).LE.VMEAS(INUM).AND.
0051 & AL(INUM,ISUB).GT.VMEAS(INUM))) GOTO 50
0052 IF(ISWITCH.EQ.0) GOTO 80
0053 CALL SDRPT(1)
0054 IOFLG=0
0055 RETURN
0056 50 IF(ISWITCH.EQ.0) GOTO 80
0057 CALL SDRPT(2)
0058 IOFLG=0
0059 RETURN
0060 60 IF(ISWITCH.EQ.0) GOTO 80
0061 CALL SDRPT(3)
0062 GOTO 80
0063 70 CALL ENABLE(0)
0064 K1=IGTSP(N)
0065 CALL ENABLE(1)
0066 DO 72 I=1,256
0067 PACK(I,K1)=0
0068 CONTINUE
0069 IF(MESSEQ.EQ.126) MESSEQ=0
0070 MESSEQ=MESSEQ+1
0071 PACK(1,K1)=0
0072 PACK(2,K1)=MESSEQ
0073 PACK(3,K1)=2
0074 PACK(4,K1)=0
0075 PACK(5,K1)=25
0076 PACK(6,K1)=ISLID
0077 PACK(7,K1)=CR
0078 PACK(8,K1)=LP
0079 ENCODE(4,74,PACK(9,K1)) ICHAN
0080 PACK(13,K1)=040
0081 ENCODE(12,76,PACK(14,K1)) VMEAS(1)
0082 PACK(26,K1)=040
0083 ENCODE(12,76,PACK(27,K1)) VMEAS(2)
0084 PACK(39,K1)=040
0085 72
0086 70
0087 70
0088 70
0089 70
0090 70
0091 70
0092 70
0093 70
0094 70
0095 70
0096 70
0097 70
0098 70
0099 70
0100 70
0101 70
0102 70
0103 70
0104 70
0105 70
0106 70
0107 70
0108 70
0109 70
0110 70
0111 70
0112 70
0113 70
0114 70
0115 70
0116 70
0117 70
0118 70
0119 70
0120 70
0121 70
0122 70
0123 70
0124 70
0125 70
0126 70
0127 70
0128 70
0129 70
0130 70
0131 70
0132 70
0133 70
0134 70
0135 70
0136 70
0137 70
0138 70
0139 70
0140 70
0141 70
0142 70
0143 70
0144 70
0145 70
0146 70
0147 70
0148 70
0149 70
0150 70
0151 70
0152 70
0153 70
0154 70
0155 70
0156 70
0157 70
0158 70
0159 70
0160 70
0161 70
0162 70
0163 70
0164 70
0165 70
0166 70
0167 70
0168 70
0169 70
0170 70
0171 70
0172 70
0173 70
0174 70
0175 70
0176 70
0177 70
0178 70
0179 70
0180 70
0181 70
0182 70
0183 70
0184 70
0185 70
0186 70
0187 70
0188 70
0189 70
0190 70
0191 70
0192 70
0193 70
0194 70
0195 70
0196 70
0197 70
0198 70
0199 70
0200 70
0201 70
0202 70
0203 70
0204 70
0205 70
0206 70
0207 70
0208 70
0209 70
0210 70
0211 70
0212 70
0213 70
0214 70
0215 70
0216 70
0217 70
0218 70
0219 70
0220 70
0221 70
0222 70
0223 70
0224 70
0225 70
0226 70
0227 70
0228 70
0229 70
0230 70
0231 70
0232 70
0233 70
0234 70
0235 70
0236 70
0237 70
0238 70
0239 70
0240 70
0241 70
0242 70
0243 70
0244 70
0245 70
0246 70
0247 70
0248 70
0249 70
0250 70
0251 70
0252 70
0253 70
0254 70
0255 70
0256 70
0257 70
0258 70
0259 70
0260 70
0261 70
0262 70
0263 70
0264 70
0265 70
0266 70
0267 70
0268 70
0269 70
0270 70
0271 70
0272 70
0273 70
0274 70
0275 70
0276 70
0277 70
0278 70
0279 70
0280 70
0281 70
0282 70
0283 70
0284 70
0285 70
0286 70
0287 70
0288 70
0289 70
0290 70
0291 70
0292 70
0293 70
0294 70
0295 70
0296 70
0297 70
0298 70
0299 70
0300 70
0301 70
0302 70
0303 70
0304 70
0305 70
0306 70
0307 70
0308 70
0309 70
0310 70
0311 70
0312 70
0313 70
0314 70
0315 70
0316 70
0317 70
0318 70
0319 70
0320 70
0321 70
0322 70
0323 70
0324 70
0325 70
0326 70
0327 70
0328 70
0329 70
0330 70
0331 70
0332 70
0333 70
0334 70
0335 70
0336 70
0337 70
0338 70
0339 70
0340 70
0341 70
0342 70
0343 70
0344 70
0345 70
0346 70
0347 70
0348 70
0349 70
0350 70
0351 70
0352 70
0353 70
0354 70
0355 70
0356 70
0357 70
0358 70
0359 70
0360 70
0361 70
0362 70
0363 70
0364 70
0365 70
0366 70
0367 70
0368 70
0369 70
0370 70
0371 70
0372 70
0373 70
0374 70
0375 70
0376 70
0377 70
0378 70
0379 70
0380 70
0381 70
0382 70
0383 70
0384 70
0385 70
0386 70
0387 70
0388 70
0389 70
0390 70
0391 70
0392 70
0393 70
0394 70
0395 70
0396 70
0397 70
0398 70
0399 70
0400 70
0401 70
0402 70
0403 70
0404 70
0405 70
0406 70
0407 70
0408 70
0409 70
0410 70
0411 70
0412 70
0413 70
0414 70
0415 70
0416 70
0417 70
0418 70
0419 70
0420 70
0421 70
0422 70
0423 70
0424 70
0425 70
0426 70
0427 70
0428 70
0429 70
0430 70
0431 70
0432 70
0433 70
0434 70
0435 70
0436 70
0437 70
0438 70
0439 70
0440 70
0441 70
0442 70
0443 70
0444 70
0445 70
0446 70
0447 70
0448 70
0449 70
0450 70
0451 70
0452 70
0453 70
0454 70
0455 70
0456 70
0457 70
0458 70
0459 70
0460 70
0461 70
0462 70
0463 70
0464 70
0465 70
0466 70
0467 70
0468 70
0469 70
0470 70
0471 70
0472 70
0473 70
0474 70
0475 70
0476 70
0477 70
0478 70
0479 70
0480 70
0481 70
0482 70
0483 70
0484 70
0485 70
0486 70
0487 70
0488 70
0489 70
0490 70
0491 70
0492 70
0493 70
0494 70
0495 70
0496 70
0497 70
0498 70
0499 70
0500 70
0501 70
0502 70
0503 70
0504 70
0505 70
0506 70
0507 70
0508 70
0509 70
0510 70
0511 70
0512 70
0513 70
0514 70
0515 70
0516 70
0517 70
0518 70
0519 70
0520 70
0521 70
0522 70
0523 70
0524 70
0525 70
0526 70
0527 70
0528 70
0529 70
0530 70
0531 70
0532 70
0533 70
0534 70
0535 70
0536 70
0537 70
0538 70
0539 70
0540 70
0541 70
0542 70
0543 70
0544 70
0545 70
0546 70
0547 70
0548 70
0549 70
0550 70
0551 70
0552 70
0553 70
0554 70
0555 70
0556 70
0557 70
0558 70
0559 70
0560 70
0561 70
0562 70
0563 70
0564 70
0565 70
0566 70
0567 70
0568 70
0569 70
0570 70
0571 70
0572 70
0573 70
0574 70
0575 70
0576 70
0577 70
0578 70
0579 70
0580 70
0581 70
0582 70
0583 70
0584 70
0585 70
0586 70
0587 70
0588 70
0589 70
0590 70
0591 70
0592 70
0593 70
0594 70
0595 70
0596 70
0597 70
0598 70
0599 70
0600 70
0601 70
0602 70
0603 70
0604 70
0605 70
0606 70
0607 70
0608 70
0609 70
0610 70
0611 70
0612 70
0613 70
0614 70
0615 70
0616 70
0617 70
0618 70
0619 70
0620 70
0621 70
0622 70
0623 70
0624 70
0625 70
0626 70
0627 70
0628 70
0629 70
0630 70
0631 70
0632 70
0633 70
0634 70
0635 70
0636 70
0637 70
0638 70
0639 70
0640 70
0641 70
0642 70
0643 70
0644 70
0645 70
0646 70
0647 70
0648 70
0649 70
0650 70
0651 70
0652 70
0653 70
0654 70
0655 70
0656 70
0657 70
0658 70
0659 70
0660 70
0661 70
0662 70
0663 70
0664 70
0665 70
0666 70
0667 70
0668 70
0669 70
0670 70
0671 70
0672 70
0673 70
0674 70
0675 70
0676 70
0677 70
0678 70
0679 70
0680 70
0681 70
0682 70
0683 70
0684 70
0685 70
0686 70
0687 70
0688 70
0689 70
0690 70
0691 70
0692 70
0693 70
0694 70
0695 70
0696 70
0697 70
0698 70
0699 70
0700 70
0701 70
0702 70
0703 70
0704 70
0705 70
0706 70
0707 70
0708 70
0709 70
0710 70
0711 70
0712 70
0713 70
0714 70
0715 70
0716 70
0717 70
0718 70
0719 70
0720 70
0721 70
0722 70
0723 70
0724 70
0725 70
0726 70
0727 70
0728 70
0729 70
0730 70
0731 70
0732 70
0733 70
0734 70
0735 70
0736 70
0737 70
0738 70
0739 70
0740 70
0741 70
0742 70
0743 70
0744 70
0745 70
0746 70
0747 70
0748 70
0749 70
0750 70
0751 70
0752 70
0753 70
0754 70
0755 70
0756 70
0757 70
0758 70
0759 70
0760 70
0761 70
0762 70
0763 70
0764 70
0765 70
0766 70
0767 70
0768 70
0769 70
0770 70
0771 70
0772 70
0773 70
0774 70
0775 70
0776 70
0777 70
0778 70
0779 70
0780 70
0781 70
0782 70
0783 70
0784 70
0785 70
0786 70
0787 70
0788 70
0789 70
0790 70
0791 70
0792 70
0793 70
0794 70
0795 70
0796 70
0797 70
0798 70
0799 70
0800 70
0801 70
0802 70
0803 70
0804 70
0805 70
0806 70
0807 70
0808 70
0809 70
0810 70
0811 70
0812 70
0813 70
0814 70
0815 70
0816 70
0817 70
0818 70
0819 70
0820 70
0821 70
0822 70
0823 70
0824 70
0825 70
0826 70
0827 70
0828 70
0829 70
0830 70
0831 70
0832 70
0833 70
0834 70
0835 70
0836 70
0837 70
0838 70
0839 70
0840 70
0841 70
0842 70
0843 70
0844 70
0845 70
0846 70
0847 70
0848 70
0849 70
0850 70
0851 70
0852 70
0853 70
0854 70
0855 70
0856 70
0857 70
0858 70
0859 70
0860 70
0861 70
0862 70
0863 70
0864 70
0865 70
0866 70
0867 70
0868 70
0869 70
0870 70
0871 70
0872 70
0873 70
0874 70
0875 70
0876 70
0877 70
0878 70
0879 70
0880 70
0881 70
0882 70
0883 70
0884 70
0885 70
0886 70
0887 70
0888 70
0889 70
0890 70
0891 70
0892 70
0893 70
0894 70
0895 70
0896 70
0897 70
0898 70
0899 70
0900 70
0901 70
0902 70
0903 70
0904 70
0905 70
0906 70
0907 70
0908 70
0909 70
0910 70
0911 70
0912 70
0913 70
0914 70
0915 70
0916 70
0917 70
0918 70
0919 70
0920 70
0921 70
0922 70
0923 70
0924 70
0925 70
0926 70
0927 70
0928 70
0929 70
0930 70
0931 70
0932 70
0933 70
0934 70
0935 70
0936 70
0937 70
0938 70
0939 70
0940 70
0941 70
0942 70
0943 70
0944 70
0945 70
0946 70
0947 70
0948 70
0949 70
0950 70
0951 70
0952 70
0953 70
0954 70
0955 70
0956 70
0957 70
0958 70
0959 70
0960 70
0961 70
0962 70
0963 70
0964 70
0965 70
0966 70
0967 70
0968 70
0969 70
0970 70
0971 70
0972 70
0973 70
0974 70
0975 70
0976 70
0977 70
0978 70
0979 70
0980 70
0981 70
0982 70
0983 70
0984 70
0985 70
0986 70
0987 70
0988 70
0989 70
0990 70
0991 70
0992 70
0993 70
0994 70
0995 70
0996 70
0997 70
0998 70
0999 70
1000 70

```

ISEND RED REPORT

ISEND AMBER REPORT

ISEND ALARM REPORT

PAGE 003

```

FORTRAN IV      V02.1-11
0088      ENCODE(12,76,PACK(40,K1)) VMEAS(3)
0089      PACK(52,K1)=040
0090      ENCODE(12,76,PACK(53,K1)) VMEAS(4)
0091      PACK(65,K1)=040
0092      ENCODE(12,76,PACK(66,K1)) VMEAS(5)
0093      PACK(78,K1)=040
0094      ENCODE(12,76,PACK(79,K1)) VMEAS(6)
0095      PACK(91,K1)=08
0096      PACK(92,K1)=1F
0097      ENCODE(12,76,PACK(93,K1)) VMEAS(7)
0098      PACK(105,K1)=040
0099      ENCODE(12,76,PACK(106,K1)) VMEAS(8)
0100      PACK(118,K1)=040
0101      ENCODE(12,76,PACK(119,K1)) VMEAS(9)
0102      PACK(131,K1)=040
0103      ENCODE(12,76,PACK(132,K1)) VMEAS(10)
0104      PACK(144,K1)=040
0105      ENCODE(12,76,PACK(145,K1)) VMEAS(11)
0106      PACK(157,K1)=040
0107      ENCODE(12,76,PACK(158,K1)) VMEAS(12)
0108      PACK(170,K1)=040
0109      ENCODE(12,76,PACK(171,K1)) VMEAS(13)
0110      PACK(183,K1)=08
0111      PACK(184,K1)=1F
0112      PACK(185,K1)=1F
0113      FORMAT(14)
0114      74      FORMAT(F12.6)
0115      CALL ENSTR(PACK(254,K1),185)
0116      CALL ENABLE(0)
0117      CALL ENQUE(PXINQ,XINQ,K1)
0118      CALL ENABLE(1)
0119      ICHNSW=2
0120      IOFLG=0
0121      RETURN
0122      END

```

```

FORTRAN IV      V02.1-11      PAGE 201

0001 SUBROUTINE SDRDT(IC)
0002 REAL*4 RH,PL,AH,AL,VMEAS
0003 LOGICAL*41 YOUTER,XINBF,ACKSEQ,LIDFD
0004 LOGICAL*41 PACK,DATA,ISLID,MONITOR,LP,PTX,CP,DUM
0005 INTEGER*2 XINC,XOUTQ,ACKQ,PXINC,PXOUTQ,PINC
0006 INTEGER*2 FWCNT,OUTFCT,OUTQ,RESNM
0007 COMMON/ECOM/ISLID,SH(13,3),EL(13,3),AH(13,3),AL(13,3),
0008      ITEM,LINE,MON,VMEAS(13),ISUB,INUM,
0009      IDATA(12),ICHNSM,ICHAN
0010 COMMON /BUIS/ XOUTER(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0011      IOFLG,LEFLG,IRSEN
0012 COMMON/MESS/MESSEQ
0013 COMMON PACK(256,64)
0014 COMMON/QUE/XINC(15),PXINC(2),XOUTQ(16),PXOUTQ(2),
0015      ACKQ(16),PACKQ(2),INC(16),PINC(2)
0016 COMMON/3LOB/ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0017      RESNM,IALTPT,ISLID,MONITOR,LP,PTX,CP,DUM
0018 CALL ENABLE(0)
0019 X1=IGETSP(N)
0020 CALL ENABLE(1)
0021 DO 10 I=1,256
0022   PACK(I,X1)=0
0023   PACK(2,X1)=MESSEQ
0024   PACK(3,X1)=0
0025   PACK(4,X1)=0
0026   PACK(5,X1)=MONITOR
0027   PACK(6,X1)=ISLID
0028   IF (IC .EQ. 3) GOTO 50
0029   DO 20 I=7,14
0030     PACK(I,X1)=XINBF(I+50)
0031   CONTINUE
0032   ENCODE(2,32,PACK(15,X1)) LINE
0033   FORMAT(A2)
0034   FORMAT(A4)
0035   PACK(17,X1)=IC
0036   ENCODE(4,40,PACK(18,X1)) VMEAS(INUM)
0037   ENCODE(2,30,PACK(22,X1)) MON
0038   PACK(24,X1)=ITEM
0039   CALL ENSTR(PACK(254,X1),24)
0040   GOTO 70
0041   DO 60 I=7,12
0042     PACK(I,X1)=X'
0043   CONTINUE
0044   DO 62 I=11,14
0045     PACK(I,X1)=XINBF(I)
0046   CONTINUE
0047   PACK(15,X1)=CP
0048   PACK(16,X1)=LP
0049   PACK(17,X1)=PTX
0050   CALL ENSTR(PACK(254,X1),17)
0051

```

PAGE 002

```
FORTRAN IV      V02.1-11
0052 70 CALL ENABLE(0)
0053 CALL ENQUE(PXING,XING,X1)
0054 CALL ENABLE(1)
0055 RETURN
0056 END
```


PAGE 021

```

FORTRAN IV      V02.1-11
0001      SUBROUTINE BWSALP
0002      LOGICAL*1 IDATA
0003      REAL*4 RH,RL,AH,AL,VMEAS
0004      COMMON/BCON/ISWCH,RH(13,3),RL(13,3),AH(13,3),AL(13,3),
      &      ITERM,LINK,MON,VMEAS(13),ISUB,INUM,
      &      IDATA(10),ICHNSW,ICHAN
0005      IF(IDATA(1).EQ."117".AND.
      &      IDATA(2).EQ."116") GOTO 50
0007      IF(IDATA(1).EQ."117".AND.
      &      IDATA(2).EQ."106".AND.
      &      IDATA(3).EQ."106") GOTO 60
0009      IF(IDATA(1).EQ."115") GOTO 70
0011      GOTO 80
0012      ISWCH=1
0013      ITERM=IDATA(4)
0014      RETURN
0015      ISWCH=0
0016      RETURN
0017      DECODE(4,75,IDATA(2),ERR=80) ICHAN
0018      FORMAT(I4)
0019      CALL SOUT(ICHAN)
0020      ICHNSW=1
0021      RETURN
0022      END

```

```

FDM-MACRO          MACRO V03.02B1-OCT-79 09:01:28 PAGE 1

1  .TITLE FDM-MACRO
2  .SETTL NODE 26
3  .IDENT /V3.0/
4  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WTOKEN,RAM,STATUS,STATB0
6  .GLOBL LPOUT,RSTART,NDPNT,DESTR,ENSTR,LPINPT
7  .NLIST CND
8  .PSECT
9
10 000000
11 000001
12 000002
13 000003
14 000004
15 000005
16 000006
17 000007
18
19
20
21
22 172410
23 172412
24 172414
25 172416
26 172416
27 177560
28 177562
29 177564
30 177566
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

;INTERFACE ADDRESS
BAR= 172410
WCR= 172412
CSR= 172414
IOBP= 172416
OPRG= 172416
HRCSE= 177560
HRCSE= 177562
HRCSE= 177564
HRCSE= 177566

;INTERFACE OPTIONS
XCIO=1
XSIO=0
XGIO=1
NDXX=0
ND24=1
NHST=0
DHST=1

;LOCAL VARIABLES
ARG1= 2
ARG2= 4
ARG3= 6
ARG4= 10
BIT15= 100000
BIT14= 40000
BIT03= 400
BIT07= 200
BIT06= 100
BIT05= 40
BIT04= 20
BIT03= 10
BIT02= 4

```

MACRO V03.02B1-OCT-79 09:01:28 PAGE 1-1

FDM-MACRO
NODE 26

BIT01= 2
BIT00= 1

00002
000001

66
67
68

102 000232	012737	001400	172416	MOV	#1400, Q#OPREG	
103 000240	105737	172414		TSTB	Q#CSR	;FALSE READ DATA
104 000244	100375			BPL	-4	;GOOD RD
105 000246	105037	172414		CLRB	Q#CSR	;NO RETRY
106 000252	012701	010510		MOV	#10510, R1	;CLEAR DONE BIT
107 000256	012700	010540		MOV	#10540, R0	;RDBUFADR COMMAND
108 000262	004767	000016		JSR	PC, ZEROBP	;SEL OUTBUF0 COMMAND
109 000266	012701	010710		MOV	#10710, R1	
110 000272	012700	010740		MOV	#10740, R0	
111 000276	004767	000002		JSR	PC, ZEROBP	
112 000302	000440			BR	STAT5	
113 000304	00137	172416		ZEROBF: MOV	R1, Q#OPREG	
114 000310	012737	001400	172416	MOV	#1400, Q#OPREG	
115 000316	105737	172414		TSTB	Q#CSR	
116 000322	100375			BPL	-4	
117 000324	105037	172414		CLRB	Q#CSR	
118 000330	013702	172416		MOV	Q#IOBUF, R2	
119 000334	042702	172400		BIC	#177400, R2	
120 000340	010037	172416		MOV	R0, Q#OPREG	
121 000344	022702	000000		CMP	#0, R2	
122 000350	001412			BEQ	35	
123 000352	012737	001400	172416	MOV	#1400, Q#OPREG	
124 000360	105737	172414		TSTB	Q#CSR	
125 000364	100375			BPL	-4	
126 000366	105037	172414		CLRB	Q#CSR	
127 000372	005302			DEC	R2	
128 000374	000763			BR	25	
129 000376	010037	172416		MOV	R0, Q#OPREG	
130 000402	000207			RTS	PC	
131						
132						
133						
134 000404	012737	010400	172416	STAT5: MOV	#4352, Q#OPREG	
135 000412	012737	002400	172416	MOV	#1280, Q#OPREG	
136 000420	012737	002400	172416	MOV	#1280, Q#OPREG	
137 000425	000207			RTS	PC	
138						
139						

FDM-MACRO
NODE 26

MACRO V03.02B1-OCT-79 09:01:28 PAGE 3-1

```

58 000722 132767 000002 000254 B1$: BITB #BIT01, CAUSE
59 000730 001423 BEQ RTI$
60 000732 012701 000400 #INP, R1
61 000736 012702 010610 #4480, R2
62 000742 012703 010640 #4512, R3
63 000746 012704 000002 #BIT01, R4
64 000752 004767 000046 PC, EMBF
65 000756 012705 001206 #AREA, R5
66 000762 012767 000001 #1, AREA
67 000770 010267 000216 R2, DATA
68 000774 004767 000000 PC, LFINPT
69
70 001000 012605 RTI$: MOV (SP)+, R5
71 001002 012604 MOV (SP)+, R4
72 001004 012603 MOV (SP)+, R3
73 001006 012602 MOV (SP)+, R2
74 001010 012601 MOV (SP)+, R1
75 001012 012600 MOV (SP)+, R0
76 001014 052737 BIS #BIT14, @#CSR
77 001022 000002 RTI
78
79 001024 012737 EMBF: MOV #4480, @#OPREG
80 001032 012737 MOV #1280, @#OPREG
81 001040 013700 MOV @#IOBUF, R0
82 001044 130400 BITB R4, R0
83 001046 001002 BNE CRCOK
84 001050 012704 MOV #-1, R4
85 001054 010237 MOV R2, @#OPREG
86 001060 012737 MOV #768, @#OPREG
87 001066 105737 TSTB @#CSR
88 001072 100375 BPL -4
89 001074 013702 MOV @#IOBUF, R2
90 001100 042702 BIC #177400, R2
91 001104 010200 MOV R2, R0
92 001106 005400 NEG R0, @#OPREG
93 001110 010037 MOV R0, @#OPREG
94 001114 010137 MOV R1, @#OPREG
95 001120 010337 MOV R3, @#OPREG
96 001124 012737 MOV #768, @#OPREG
97 001132 105737 TSTB @#CSR
98 001136 100375 BPL -4
99 001140 012737 MOV #8704, @#OPREG
100 001146 002240 NOP
101 001150 105737 TSTB @#CSR
102 001154 102421 EMI DMAOK
103 001156 000240 NOP
104 001160 012737 MOV #2304, @#OPREG
105 001166 105737 TSTB @#CSR
106 001172 100375 BPL -4
107 001174 005704 TST R4
108 001176 100001 BPL END$
109 001200 010402 MOV R4, R2
110 001202 000207 RTS PC
111 001204 000000 CAUSE: .WORD 0
112 001206 000000 AREA: .WORD 0
113 001210 001212 .WORD DATA
114 001212 000000 DATA: .WORD 0

```

```

;INBUF1 FULL
;NO MORE DONE
;BUFFER ADDRESS
;RDBUFADR COMM
;SEL INBUF1
;CRC BIT
;GO EMPTY BUFFER
;DATA LINK AREA
;ONE VARIABLE
;CRC OF BYTE COUNT
;CALL FORTRAN QUE'ER
;
;RESTORE REGISTERS

```

```

;ENABLE INTERRUPTS
;RETURN FROM INTERRUPT

```

```

;READ STATUS 1
;RS
;FETCH STATUS
;GOOD CRC

```

```

;NO FLAG
;RDBUFADR
;RD
;GOOD RD

```

```

;REPLACE WITH POINTER
;CLEAR MST BITS
;SAVE IT
;2'S COMP

```

```

;BYTE COUNT NOW
;ADDRESS IN MEMORY
;SEL BUFFER
;FALSE RD

```

```

;DONE ON
;NO LOOP
;FIRE DMA
;DELAY
;GOOD DMA

```

```

;ERROR IF HERE
;FALSE WD
;GOOD WD

```

```

;WAS CRC OK
;YES
;NO FLAG IT

```

```

;RETURN
;STA US BYTE 0 HOLDER

```

EDM MACRO
NODE 26

MACRO V03.02B1-OCT-79 09:01:28 PAGE 4

```

1      .ENABLE LSR
2      *****
3      ***** PROCEDURE ENABLE *****
4      *****
5
6      001214 017501 000002      ENSTR: MOV  GARG1(R5), R1
7      001220 022701 000001      ENSTR: CMP  #1,      R1
8      001224 001404      ENSTR: BEQ  E1$,      @#CSR
9      001226 042737 040000 172414  ENSTR: BIC  #BIT14, @#CSR
10     001234 000207      ENSTR: RTS
11     001236 052737 040000 172414  ENSTR: BIS  #BIT14, @#CSR
12     001244 000207      ENSTR: RTS
13
14      *****
15      ***** PROCEDURE LFCODE MACRO *****
16     001246 017575 000004 000002  ENSTR: MOV  GARG2(R5), GARG1(R5)
17     001254 000207      ENSTR: RTS
18     001256 017575 000002 000004  ENSTR: MOV  GARG1(R5), GARG2(R5)
19     001264 000207      ENSTR: RTS
20
21      *****
22      ***** PROCEDURE LINE SWITCH *****
23      *****
24      *****
25
26      *****
27      *****
28      *****
29      *****
30      *****
31      *****
32
33     001266 017501 000002      SWITCH: MOV  GARG1(R5), R1
34     001272 012737 010420 172416  SWITCH: MOV  #4368., @#OPREG
35     001300 062701 004400      SWITCH: ADD  #2304., R1
36     001304 010137 172416      SWITCH: MOV  R1,      @#OPREG
37     001310 105737 172414      SWITCH: TSTB  @#CSR
38     001314 100375      SWITCH: BPL  .-4
39     001316 105037 172414      SWITCH: CLRB  @#CSR
40     001322 000207      SWITCH: RTS
41
42

```

; WHICH SUBROUT.

; DISABLE LIU INTERRUPTS

; ENABLE LIU

; ENCODE BYTE COUNT

; DECODE BYTE COUNT

; CALL SWITCH(X) -SET OR RESET LINE SWITCHS

; 04=SET PRIMARY LINE

; 32=RESET PRIMARY LINE

; 08=SET BACKUP LINE

; 64=RESET BACKUP LINE

; SW SET

; WCR : MODSTAT

; RD

; VALID WRITE

; NO LOOP UNTIL READY

; CLEAR DONE BIT

EDM.MACHO
NODE 26

MACRO V03.02B1-OCT-79 09:01:28 PAGE 6

[illegible]

EDM.MACRO
NODE 26

MACRO V03.02B1-OCT-79 09:01:28 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9          ;CALL RSTART(0) -CAUSES A SOFTWARE HALT
10         ;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
11         ;CALL RSTART(2) -LOAD MODE(173000)
12
13         *****
14         *****
15         *****
16         *****
17         *****
18         *****
19         *****
20         *****
21         *****
22         *****
23         *****
24
25
26

```

;MODE
 ;LOAD ADDRESS
 ;RESTART PROGRAM
 ;INT LIU FIRST
 ;START ADDRESS
 ;LOAD MODE

RSTART:	MOV	GARG1(R5), R0			
RS0:	COMP	#0,	R0		
	BNE	RS1			
	HALT				
	RTS	PC			
RS1:	COMP	#1,	R0		
	BNE	RS2			
	JSR	PC,	LIUINT		
G4\$:	MOV	#40,	R0		
	JMP	(R0)			
RS2:	COMP	#2,	R0		
	BNE	RS1			
	JMP	GROM			
RST:	RTS	PC			
ROM:	.WORD	173000			

MACRO V03.02B1-OCT-79 09:01:28 PAGE 11

```

1 *****
2 .ENABLE LSB *****
3 *****
4 *****
5 *****
6 SIO:: MOV R0, -(SP) ;SAVE REGISTERS
7 MOV R1, -(SP)
8 MOV R2, #XINEF, R0 ;BUFFER ADDRESS
9 MOV #80., R1 ;BYTE COUNT
10 TSTB Q#HRCR ;PORT READY
11 BPL -4 ;LOOP UNTIL
12 MOVB Q#HRBUF, R2 ;FETCH BYTE
13 BIC #177400, R2 ;CLEAR MST
14 MOV IOFLG, R2 ;ITS THE BYTE COUNT
15 MOVB R2, (R0)+ ;COUNT-1
16 DEC R1 ;LOOP READY
17 TSTB Q#HRCR ;PORT UNTIL
18 BPL -4 ;FETCH BYTE
19 MOVB Q#HRBUF, R2 ;FETCH BYTE
20 BIC #177400, R2 ;CLEAR MST
21 MOVB R2, (R0)+ ;STORE IT
22 DEC R1
23 BNE 25 ;READ 39 BYTES
24 BIC #100, Q#HRCR ;NO MORE UNTIL FORTRAN
25 MOV (SP)+, R2 ;RETORE REGISTERS
26 MOV (SP)+, R1
27 MOV (SP)+, R0
28 RTI
29
30 ;SEND OVER TO SIG
31
32 SOUT:: MOV GARG1(R5), R0 ;BYTE COUNT
33 TSTB Q#HXCSR ;READY TO SNED
34 BPL -4 ;LOOP UNTIL
35 MOVB R0, Q#HXBUF ;SEND IT
36 RTS
37
38 ;REPORT MODE READY
39
40 READY:: TSTB Q#HXCSR ;PORT READY
41 BPL -4 ;LOOP UNTIL
42 MOVB #122, Q#HXBUF ;SEND R
43 BIS #100, Q#HRCR ;ENABLE INTERRUPTS
44 RTS ;RETURN

```


FDM.MACRO
NODE 26

MACRO V03.02B1-OCT-79 09:01:28 PAGE 12

```
*****  
*****  
***** COMMON DATA AREA *****  
*****  
*****  
PSECT DFM,RW,D,CBL,REL,OVR  
  
OUTBF: .BLKB 255.  
INBF: .BLKB 256.  
IWRITTN: .BLKW  
IWRT: .BLKW  
STAT: .BLKW  
LTIME: .BLKW  
.EVEN  
  
PSECT BUFS,RW,D,CBL,REL,OVR  
  
XOUTBF: .BLKB 256.  
XINBF: .BLKB 256.  
ACKSEQ: .BLKB 256.  
LIDFD: .BLKB 256.  
IOFLG: .BLKW  
LIPLG: .BLKW  
IASEND: .BLKW  
.EVEN  
  
END
```

000001

FDM MACRO SYMBOL TABLE

MACRO 003.01 23-JUL-79 12:51:52 PAGE 12-1

ACKSED	001000R	003	H1\$	000222R	E1\$	001236R	MASTER	000000RG	SOUT	002576RG	002
AREA	001206R		H2\$	000510R	64\$	002444R	NIXX =	000000	STAT	001004R	
ARG1	= 000002		B3\$	000534R	HRBUF =	177562	NIX4 =	000001	STAT0	001422RG	
ARG2	= 000004		B4\$	000560R	HRCSR =	177560	NH051 =	000000	STATUS	001324RG	
ARG3	= 000006		H5\$	000606R	HXRUF =	177566	0FREG =	172416	STAT\$	000404R	
ARG4	= 000010		B6\$	000624R	HXC5R =	177564	OUTBF	000000R	SWITCH	001266RG	
BAR	= 172410		B7\$	000570R	INBF	000400R	002	EAM	TIME	001654RG	
BIT00	= 000001		CAUSE	001204R	IOHUF =	172416	003	KDFNT	WCR =	172412	
BIT01	= 000002		CRCON	001054R	IOFLG	002000R	003	KIRAM	WTH	001652R	
BIT02	= 000004		CSR	= 172414	IRSEND	002004R	003	READY	WTKG	001552R	
BIT03	= 000010		DATA	001212R	IMRT	001002R	002	ROM	WTKEN	001474RG	
BIT04	= 000020		DESTR	001256RG	IMRTTH	001000R	002	RST	WTRAM	001710R	
BIT05	= 000040		DH0ST =	000001	LIHFD	001400R	003	RSTART	XCIO =	000001	
BIT06	= 000100		DMAOK	001160R	LIO	000430RG		RSO	XGIO =	000001	
BIT07	= 000200		EMBF	001024R	LIJINT	000076RG		RS1	XINBF	000400R	003
BIT08	= 000400		ENABLE	001214RG	LIFLG	002002R	003	RS2	XOUTBF	000000R	003
BIT14	= 040000		END\$	001202R	LFINFT =	***** G		RS2	XSID =	000000	
BIT15	= 100000		ENSTR	001246RG	LFOUT	002102RG		RS1\$	ZEROBF	000304R	
B0\$	000644R		E0\$	001226R	LTIME	001006R		SIO			

* ABS. 000000 000
002642 001
IDM 001010 002
RUF5 002006 003
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 51 PAGES
*DK:FDM=DK:FDM

PAGE 001

12:00:00

01-JUL-79

CHINA, S. C. CHINA

POSTAL

DESCRIPTION

FOR RUSSIA, NOT THE
FOR RUSSIA, NOT THE
FOR RUSSIA, NOT THE
FOR RUSSIA, NOT THE
FOR RUSSIA, NOT THE

Page 001

12:00:00

01-JUL-79

100.00.000

PROGRAM:

ALCUTAL

ASSIGN DX0: DN:

K LINE

DX1:R00026.DX1:R00P=DX1:R0001 ZC/LM

DX1:R0001 ZC

DX1:R0001 ZC

DX1:R0001 ZC

\$SMR1

ASSIGN DX1: DN:

1.8 NODE 27 (FIAC)

The Fault Isolation and Control Coordination (FIAC) module interprets Event Reports and Alarms from the VSQC, DSQC, and BWBSA modules for the purpose of isolating the equipment causing the detection of a fault condition. Amber and Red Event Reports are received by the FIAC and retransmitted to the destination Node Designator for OCRI reporting and to the DBMS in order to update the Equipment Status File. The OCRI operator can display and modify the Equipment Status File via Mode 6 of the User Language. Red Event Reports are reported only once when the equipment first fails although subsequent measurements will also result in the Red Region.

The FIAC analyzes the Red Event Reports for fault indications and initiates isolation procedures to resolve to the equipment level the location of the fault. The FIAC checks the connectivity through the network. For the FDM, the assumed connectivity between the VSQC, DSQC and BWBSDA is:

FDM Assumed Connectivity

<u>Connectivity Group</u>	<u>Monitor Points</u>
Area 1	1-333
Area 2	334-666
Area 3	667-1000

The FIAC collects Red Event Reports and assigns them to the various connectivity groups. When a connectivity group has a Red Event Report from VSQC, DSQC and BWBSA, the monitor points are compared to determine the equipment causing the fault. The lowest monitor point indicates the equipment causing the fault. Fault Reports are then sent to both the local OCRI terminal and a remote site OCRI. The Fault Report contains the monitor points, link and channel numbers for the connectivity group. The FIAC software contains variables used to specify the Node Designators for the Local and Remote destinations for Fault Reports. The default nodes are the OCRI terminal for Local Fault Reports and the CRT terminal in loop 4 for Remote Fault Reports.

In the MSCDM, software resides on the SDCA node to simulate the behavior of a remote FIAC. Area 1 is used to represent the area associated with the MSCDM FIAC. Area 2 is used to represent the area associated with the remote FIAC (as simulated by SDCA). Area 3 is used to simulate other areas for which faults cannot be isolated by the FIAC. The SDCA sends occasional random Event Reports to FIAC with Monitor Points in Area 1. FIAC sends Event Reports to SDCA for those Event Reports received from VSQC, DSQC, and BWBSA with Monitor Points in Area 2. When Red Event Reports are collected for the VSQC, DSQC, and BWBSA, faults can be isolated and reported in Areas 1 and 2 but cannot be isolated for Area 3. The faults that cannot be isolated are reported to local and remote OCRI's via a Fault Report.

1.8.1 Program Descriptions

1.8.1.1 Refer to Section 1.1 for descriptions of NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME, RAM, LPOUT, RDPNT, RSTART.

1.8.1.2 Subroutine FIAC (FORTRAN)

This subroutine is called when a message is received from the loop and perform the functions described in Section 1.8.

1.8.1.3 Subroutine SNDRPT (FORTRAN)

The subroutine is called by FIAC routine to send event reports to DBMS.

1.8.1.4 Subroutine FAULT (FORTRAN)

This subroutine performs the fault isolation described in Section 1.8.

```

PAGE 002
0001  PROGRAM MODAL
0002  INTEGER*2 XI(16),XOUTQ,ACXQ,PXIN, PXOUTQ,PACKQ
0003  INTEGER*2 P1,Q1,PRER,SAT,ILCMN,OUTPT
0004  INTEGER*2 SEQR,SESTP,STEEP
0005  INTEGER*2 RSTXP,Q1,Q,REXLM,OUTQ,LEQR
0006  LOGICAL*1 XOUTP,XINP,ACKSEQ,LLFD,OUTP,LEF
0007  LOGICAL*1 IDTA,RESD
0008  LOGICAL*1 PA,EA,CR,LP,MONTP,ISLD,DUM
0009  LOGICAL*1 LTIME,ACKTM,NETIM,OLDP,INTIM,TIMM,ATIMM
0010  REAL*4 FPEAS
0011  REAL*8 FZ
0012  COMMON /MISS/ MESSIQ
0013  COMMON /DM/ OUTP(256),LVEF(256),
0014  COMMON /INTIM/ INT,STAT,LTIME
0015  1  COMMON /QUT/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0016  ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0017  1  COMMON /BUS/ XOUTP(256),XINP(256),ACKSEQ(256),
0018  LLFD(256),TOPLO,LLFLG,IRSEND
0019  1  COMMON /PR/ PRER(64),IFR,FRSZ
0020  COMMON /TIM/ OLDTIM,TIMM,ACKTM,ATIMM
0021  COMMON /SWT/ SETPR,SEPRM,SEPRP,RSTBP
0022  COMMON /GLOB/ ISNT,PWCN,ICNTH,OUTPT,IFULL,OUTQ,
0023  ASSLM,IALERT,ISLD,MONTP,IF,ETI,CE,DUM
0024  1  COMMON /SCOM/ ICNTH(32),DATA(80),ICNTH,MON,PR,FPEAS,
0025  ITYPE,IARZEA,ITPRM,PRD(12,4)
0026  CALL MASTER
0027  CALL INIT
0028  CALL ENBLE(1)
0029  5  CONTINUE
0030  25  IF(PINQ(1).LT. PINQ(2)) GOTO 40
0031  CALL ENBLE(2)
0032  Q2=DEQUE(PINQ,INQ,1)
0033  CALL ENBLE(1)
0034  DO 30 I=1,82
0035  1  DATA(1)=PACK(I,Q2)
0036  CONTINUE
0037  CALL FLAC
0038  IPR=IPR+1
0039  PRD(IFR)=Q2
0040  CALL STATP(15)
0041  IF(Q1.EQ.1) GOTO 100
0042  IF(LLFLG.GT.2) CALL LINLOS
0043  IF(OUTPT.EQ.1) GOTO 100
0044  IF(IFULL.EQ.1).OR.(ISNT.EQ.1) GOTO 100
0045  IF(IRSEND.EQ.1) GOTO 85
0046  IF(PXINQ(1).LT. PXINQ(2)) GOTO 100
0047  CALL ENBLE(2)
0048  Q1=DEQUE(PXINQ,XINQ,1)
0049  CALL ENBLE(1)
0050  OUTQ=Q1
0051  DO 50 I=1,Q2
0052  CALL DESTR(PACK(254,OUTQ),Q2)
0053  OUTP(I)=PACK(I,OUTQ)

```

PAGE 222

```

FORTRAN IV      V02.1-11
2057  90  CONTINUE
2058      OUTP(Q2+1)=2
2059      IP=OUTP(5)
2060      OUTP(Q2+2)=LILP(IP)
2061      IFULL=1
2062      ISENT=1
2063      CALL LPUT(Q2+2)
2064      ISEND=1
2065      ACTIM=0
2066      IRETH=0
2067      IRT=1
2068      INLCI=0
2069      OLDTIM=NEWTIM
2070      NEWTIM=TIME
2071      IF(IERT.EQ.0) GOTO 120
2072      IRETH=IRETH+(NEWTIM-OLDTIM)
2073      IF((IRETH.LT.TIMLIM) GOTO 120
2074      CALL WOKEN
2075      IRETH=0
2076      IRT=0
2077      IF(ISENT.EQ.0) GOTO 130
2078      ACTIM=ACTIM+(NEWTIM-OLDTIM)
2079      IF(ACTIM.LT.ATIMLIM) GOTO 130
2080      CALL ENAB(0)
2081      CALL ACNAR(0)
2082      CALL ENAB(1)
2083      CONTINUE
2084      GOTO 5
2085  END
2086
2087  130  CONTINUE
2088
2089

```

```

FORTRAN IV      V02.1-11      PAGE 001

0001      FUNCTION ICETSP(N)
0002      LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0003      INTEGER*2 FREE,FLCNT
0004      INTEGER*2 OUTPCT,OUTQ,RESNLM
0005      COMMON /FRE/ FREE(G4),IFR,IFRSZ
0006      COMMON /GLOB/ ISENT,FLCNT,IQLNTH,OUTPCT,IFULL,OUTQ,
0007      1 RESNLM,I,ALTRT,ISLID,MONITOR,IF,ETX,CR,DUM
0008      IF(IFR.LT.1) CALL INIT
0009      ICETSP=FREE(IFR)
0010      IFR=IFR-1
0011      RETURN
0012      END

```


PAGE 021

FORTRAN IV 022.1-11

```

0001 SUBROUTINE ENQUE(A,B,N)
0002 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003 INTEGER*2 XINQ,XOUTQ,XOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWCT,OUTQ,A(2),B(16)
0005 INTEGER*2 RESNM,OUTPCT
0006 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007 1 COMMON /GLOS/ ISENT,FLWCT,IQLENTH,OUTPCT,I*FULL,OUTQ,
0008 1 RESNM,IALTRT,ISLID,MONTOR,IF,ETX,CR,DUM
0009 10 IQHEAD=A(1)
0010 IQTAIL=A(2)
0011 IF(IQTAIL.EQ.1) GOTO 20
0012 IQTAIL=IQTAIL-1
0013 B(IQTAIL)=N
0014 A(2)=IQTAIL
0015 GOTO 999
0016 20 IF(IQHEAD.GE.(IQLENTH)) GOTO 40
0017 NN=IQHEAD-IQTAIL
0018 DO 30 I=1,NN+1
0019 B(IQLENTH+1-I)=B(IQHEAD+1-I)
0020 30 B(IQLENTH+1-I)=B(IQHEAD+1-I)
0021 A(1)=IQLENTH
0022 A(2)=IQLENTH-NN
0023 GOTO 10
0024 40 CALL INIT
0025 999 RETURN
0026 END

```

PAGE 001

FORTRAN IV V02.1-11

```

0001 FUNCTION DEQUE(A,B,LD)
0002 LOGICAL*1 ETX,CH,LF,DUM,ISLID,MONTOR
0003 INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWNT,A(2),R(16),DEQUE,OUTFCT,OUTQ,RESNLM
0005 COMMON /QUE/ XINQ(16),PXINQ(2),YOUTQ(16),PXOUTQ(2),
0006          ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0007 COMMON /GLOB/ ISENT,FLWNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008          RESNLM,IALTRT,ISLID,MONTOR,LF,ETY,CH,DUM
0009 IQHEAD=A(1)
0010 DEQUE=B(IQHEAD)
0011 IF(ID.NE.1) GOTO 999
0012 IF(IQHEAD.NE.0) GOTO 10
0013 A(1)=IQLNTH
0014 A(2)=IQLNTH+1
0015 GOTO 999
0016 10 A(1)=IQHEAD-1
0017 999 RETURN
0018 END

```

PAGE 001

XOFORTAN IV V02.1-11

```

0001 SUBROUTINE ACKNAK(N)
0002 INTEGER*2 FLWCNT,OUTFCT,STAT,XINQ,PXINQ,PXOUTQ
0003 INTEGER*2 ACKQ,PACKQ,PINQ,FREF,OUTQ,XOUTQ
0004 INTEGER*2 T1,T2,T3,T4,T5,RESNLM
0005 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0006 LOGICAL*1 PACK,OUTF,INBF,LCONT(40)
0007 LOGICAL*1 YOUTBF,XINBF,ACKSEQ,LIDFD
0008 INTEGER*2 LTIME,IWRTM
0009 REAL*8 COMT(5)
0010 EQUIVALENCE(COMT,LCONT)
0011 DATA COMT(1)/'MESSAGE',/COMT(2)/'NOT SENT',/COMT(3)/' FROM '/,
0012 1 COMT(4)/'NODE',/COMT(5)/' TO NODE',
0013 1 COMMON /GLOB/ ISENT,FLWCNT,IQNTN,OUTFCT,IFULL,OUTQ,
0014 1 RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0015 1 COMMON /DFM/ OUTBF(256),INBF(256),
0016 1 IWRTM,IWRT,STAT,LTIME
0017 1 COMMON /ACK/ PACK(256,64)
0018 1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),ACKQ(16),
0019 1 PACKQ(2),INQ(16),PINQ(2)
0020 1 COMMON /FREF/ FREF(64),IFR,IFRSZ
0021 1 COMMON /MESS/ MESSEQ
0022 1 COMMON /BUFS/ YOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0023 1 IOFLG,LLFLG,IIRSEND
0024 IF (N.NE.1) GOTO 10
0025 IFR=IFR+1
0026 DO 5 I=1,256
0027 PACK(I,OUTQ)=0
0028 IFULL=0
0029 IFULL=0
0030 ISEND=0
0031 FREE(IFR)=OUTQ
0032 GOTO 999
0033 10 T2=PACK(256,OUTQ)
0034 T2=T2+1
0035 IF (T2.LE.RESNLM) GOTO 120
0036 T3=PACK(3,OUTQ)
0037 T4=MOD(T3,64)
0038 IF (T4.LT.32) GOTO 110
0039 T5=IGETSP(N)
0040 IF (MESSEQ.EQ.126) MESSEQ=0
0041 MESSEQ=MESSEQ+1
0042 PACK(1,T5)=0
0043 PACK(2,T5)=MESSEQ
0044 PACK(3,T5)=0
0045 PACK(4,T5)=0
0046 PACK(5,T5)=25
0047 PACK(6,T5)=ISLID
0048 DO 20 I=7,60
0049 PACK(I,T5)="040
0050 20 CONTINUE
0051 DO 30 I=7,9
0052 PACK(I,T5)=LF
0053 30 CONTINUE

```

PAGE 002

```

FORTRAN IV      V02.1-11
0054      DO 40 I=1,8
0055          PACK(I+9,T5)=LCOMT(I)
0056      40  CONTINUE
0057      DO 50 I=3,16
0058          PACK(I+9,T5)=LCOMT(I)
0059      50  CONTINUE
0060      DO 60 I=17,22
0061          PACK(I+9,T5)=LCOMT(I)
0062      60  CONTINUE
0063      DO 70 I=25,29
0064          PACK(I+7,T5)=LCOMT(I)
0065      70  CONTINUE
0066      ENCODE(3,80,PACK(38,T5)) ISLID
0067      80  FORMAT(13)
0068      DO 90 I=33,40
0069          PACK(I+11,T5)=LCOMT(I)
0070      90  CONTINUE
0071      ENCODE(3,80,PACK(52,T5))PACK(5,OUTQ)
0072      PACK(55,T5)=LF
0073      PACK(56,T5)=LF
0074      PACK(57,T5)=CR
0075      PACK(58,T5)=ETX
0076      CALL ENSTR(PACK(254,T5),58)
0077      CALL ENQUE(PXINQ,XINQ,T5)
0078      DO 102 I=1,256
0079          PACK(I,OUTQ)=0
0080      102 CONTINUE
0081      ISENT=0
0082      IREND=0
0083      IFULL=0
0084      IWRT=0
0085      IFR=IFR+1
0086      FREE(IFR)=OUTQ
0087      GOTO 999
0088      110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089          PACK(256,OUTQ)=0
0090      ISENT=0
0091      IFULL=0
0092      IF(IASEND.EQ.1) GOTO 999
0093      CALL ENQUE(PXINQ,XINQ,OUTQ)
0094      GOTO 999
0095      120 PACK(256,OUTQ)=T2
0096          IFULL=0
0097      ISENT=0
0098      IF(IASEND.EQ.1) GOTO 999
0099      CALL ENQUE(PXINQ,XINQ,OUTQ)
0100      999  RETURN
0101      END
0102
0103

```

PAGE 201

FORTRAN IV V02.1-11

```

0001 SUBROUTINE INPTQ(L)
0002 INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ
0003 INTEGER*2 PACKQ,PINQ,FLWCNT,OUTFCT,T1,RESNLM,OUTQ
0004 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0005 LOGICAL*1 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
0006 INTEGER*2 LTIME,IWRITM
0007 COMMON /DFM/ OUTBF(256),INBF(256),
0008 1 IWRITM,IWR,STAT,LTIME
0009 COMMON /BUPS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010 1 LIDFD(256),IOFLG,LLFLG,IRSEND
0011 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012 1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0013 COMMON /GLOB/ ISENT,FLWCNT,IQNTHT,OUTFCT,IFULL,OUTQ,
0014 1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0015 IF(L.LT. 1) GOTO 30
0016 LI=L-2
0017 II=INBF(6)
0018 IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
0019 T1=IGETSP(N)
0020 DO 10 I=1,LI
0021 10 PACK(I,T1)=INBF(I)
0022 CALL ENSTR(PACK(254,T1),LI)
0023 CALL ENQUE(PINQ,INQ,T1)
0024 N=2
0025 GOTO 40
0026 20 N=1
0027 GOTO 40
0028 30 N=0
0029 40 DO 50 I=1,2
0030 50 OUTBF(I)=INBF(I)
0031 NN=1
0032 IF(N.EQ. 0) NN=128
0033 OUTBF(3)=NN
0034 OUTBF(4)=0
0035 OUTBF(5)=INBF(6)
0036 OUTBF(6)=ISLID
0037 OUTBF(7)=ETX
0038 OUTBF(8)=0
0039 OUTBF(9)=LIDFD(INBF(6))
0040 CALL LPOUT(9)
0041 IF(N.NE. 2) GOTO 999
0042 ACKSEQ(II)=INBF(2)
0043 999 RETURN
0044 0045 END

```



```

FORTRAN IV      V02.1-11      PAGE 001

0001      SUBROUTINE LPINPT(LI)
0002      INTEGER*2 STAT,FLWNT,OUTFCT,OUTC,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONTR,ISLID,DUM
0005      INTEGER*2 LTIME,IWRTM
0006      COMMON/DFM/ OUTBF(256),INBF(256),IWRTM,IWRT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,ELCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
1      RESNLM,IALTHT,ISLID,MONTR,LF,ETX,CR,DUM
0008      CC1=INBF(3)
0009      CC2=INBF(4)
0010      IF(CCC1 .LT. 0) GOTO 25
0011      IF(CCC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0012      IF(CCC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0013      CC3=MOD(CCC1,2)
0014      CC4=MOD(CCC1,256)
0015      IF ((CC3 .GE. 1 .OR. CC4 .GE. 128) .AND. ISENT .EQ. 1) GOTO 20
0016      IF ((INBF(1) .EQ. 85) .AND. INBF(2) .EQ. 170) GOTO 99
0017      IF ((INBF(1) .EQ. 85) .AND. INBF(2) .EQ. 170) GOTO 99
0018      CC5=MOD(INBF(3),54)
0019      IF(CCC5 .GE. 32) CALL INPTQ(LI)
0020      GOTO 99
0021      20      CC7=MOD(CCC1,2)
0022      IF(CCC7 .GE. 1) GOTO 30
0023      CALL ACKNAK(0)
0024      GOTO 99
0025      30      CALL ACKNAK(1)
0026      GOTO 99
0027      40      CALL INPTQ(LI)
0028      CONTINUE
0029      RETURN
0030      END
0031
0032
0033
0034

```

PAGE 001

FORTRAN IV V02.1-11

```

0001 SUBROUTINE INIT
0002 REAL*4 FMEAS
0003 REAL*8 FTR
0004 INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ,FREE
0005 INTEGER*2 STAT,FLCNT,OUTFCT
0006 INTEGER*2 SETPRM,RSTPRM,SETRP,OUTQ,RESNLM,RSTXP
0007 LOGICAL*1 DUM,DATA,PRED
0008 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDED,OUTBF,INBF
0009 LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID
0010 INTEGER*2 LTIME,OLTIM,TIMLM,ACKTIM,ATIMLM,IWRTTM
0011 COMMON /DFM/ OUTBF(256),INBF(256)
0012 1 IWRTTM,IWRT,STAT,LTIME
0013 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0014 1 ACKQ(16),PACKQ(2),INC(16),PINQ(2)
0015 COMMON /MESS/ MESSEQ
0016 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIDED(256),
0017 1 IOFLG,LLFLG,IRSEND
0018 COMMON /FRE/ FREE(64),IFR,IFRSZ
0019 COMMON /TIM/ OLDTIM,TIMLM,ACKTIM,ATIMLM
0020 COMMON /PACK/ PACK(256,64)
0021 COMMON /SWT/ SETPRM,RSTPRM,SETRP,RSTXP
0022 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0023 1 RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0024 COMMON /FCOM/ ICONN(3,6),IDATA(80),ICHAN,MON,FTR,FMEAS,
0025 5 ITYPE,IAREA,ITERM,PRED(1014)
0026 CALL RAM(0,9,4)
0027 CALL RAM(0,255,0)
0028 MESSEQ=0
0029 IOFLG=0
0030 IRSEND=0
0031 LLFLG=0
0032 IFRSZ=64
0033 INFLCT=0
0034 ISLID=27
0035 CR='015'
0036 LF='012'
0037 ETX='003'
0038 ICHNSW=0
0039 RESNLM=2
0040 IQLNTH=16
0041 IALTRT=0
0042 FLWCNT=0
0043 IWRTTM=0
0044 TIMLM=35
0045 SETPRM=4
0046 RSTPRM=32
0047 SETBP=8
0048 RSTBP=64
0049 ATIMLM=200
0050 MONTOR=25
0051 IWRT=0
0052 ISWICH=0
0053 OUTFCT=0
0054 IFULL=0

```

PAGE 002

```

FORTRAN IV      V02.1-11
0050      ISENT=0
0051      DO 40 I=1,20
0052      40      LIDF(I)=4
0053      LIDF(21)=1
0054      LIDF(22)=3
0055      LIDF(23)=6
0056      LIDF(24)=5
0057      LIDF(25)=7
0058      LIDF(26)=8
0059      LIDF(27)=9
0060      LIDF(28)=2
0061      DO 50 I=29,39
0062      50      LIDF(I)=0
0063      DO 60 I=40,44
0064      60      LIDF(I)=1
0065      DO 70 I=45,59
0066      70      LIDF(I)=0
0067      DO 80 I=60,64
0068      80      LIDF(I)=2
0069      DO 90 I=65,79
0070      90      LIDF(I)=0
0071      DO 100 I=80,84
0072      100     LIDF(I)=4
0073      DO 110 I=85,99
0074      110     LIDF(I)=0
0075      DO 120 I=100,104
0076      120     LIDF(I)=5
0077      DO 130 I=105,256
0078      130     LIDF(I)=0
0079      PINQ(1)=IQLNTH
0080      PINQ(2)=IQLNTH+1
0081      PACKQ(1)=IQLNTH
0082      PACKQ(2)=IQLNTH+1
0083      POUTQ(1)=IQLNTH
0084      POUTQ(2)=IQLNTH+1
0085      PXINQ(1)=IQLNTH
0086      PXINQ(2)=IQLNTH+1
0087      DO 10 I=1,IFRSZ
0088      10      PACK(255,I)=0
0089      DO 20 I=1,IFRSZ
0090      20      FREE(I)=I
0091      DO 30 I=1,256
0092      30      ACKSEQ(I)=256
0093      IFR=IFRSZ
0094      RETURN
0095      END
0096

```

PAGE 001

FORTRAN IV V02.1-11

```

0001 SUBROUTINE LINLOS
0002 REAL*8 RMI(5),LIN08,LIN18
0003 INTEGER*2 FIMCNT,OUTCT,OUTQ,RESNLM,TI
0004 INTEGER*2 XINQ,XINQ,XOUTC,PXOUT,ACKQ,PACKQ,PINQ
0005 LOGICAL*1 YOUTBF,XINBF,ACKSEQ,LIFED
0006 LOGICAL*1 ETX,CR,LF,MONTR,ISLID,DUM
0007 LOGICAL*1 PACK,M1(40),LIN0(8),LIN1(8)
0008 COMMON/QUE/XINQ(16),PXINQ(2),XOUTC(16),PXOUTQ(2),ACKQ(16),
0009 & PACKQ(2),INQ(16),PINQ(2)
0010 & COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIFD(256),
0011 & IOFIG,LLFLG,IRSEND
0012 COMMON /GLOB/ISENT,FIMCNT,IOLNTH,OUTCT,IPULL,OUTQ,
0013 & RESNLM,IATRT,ISLID,MONTR,IF,ETX,CR,DUM
0014 COMMON /MESS/ MESSEQ
0015 COMMON /PACK/ PACK(256,64)
0016 DATA LIN08,LIN18/'PRIMARY','BACKUP'//
0017 DATA RMI(1),RMI(2)/'LOSS OF','MODULATI'//
0018 DATA RMI(3),RMI(4)/'ON ON LO','OF AT NO'//
0019 DATA RMI(5)/'DE 27'//
0020 EQUIVALENCE(LIN0,LIN08)
0021 EQUIVALENCE(LIN1,LIN18)
0022 IS=LLFLG
0023 CALL ENABLE(0)
0024 TI=ICETSP(N)
0025 CALL ENABLF(1)
0026 PACK(1,TI)=0
0027 IF(MESSEQ.EQ.126) MESSEQ=0
0028 MESSEQ=MESSEQ+1
0029 PACK(2,TI)=MESSEQ
0030 PACK(3,TI)=0
0031 PACK(4,TI)=0
0032 PACK(5,TI)=25
0033 PACK(6,TI)=ISLID
0034 DO 20 I=7,9
0035 PACK(I,TI)=LF
0036 CONTINUE
0037 DO 30 I=1,22
0038 PACK(I+9,TI)=M1(I)
0039 IF(IS.EQ.1) GOTO 50
0040 DO 40 I=1,8
0041 PACK(I+31,TI)=LIN0(I)
0042 GOTO 70
0043 DO 50 I=1,8
0044 PACK(I+31,TI)=LIN1(I)
0045 DO 80 I=23,40
0046 PACK(I+17,TI)=M1(I)
0047 PACK(58,TI)=CR
0048 PACK(59,TI)=LF
0049 PACK(60,TI)=ETX
0050 CALL ENSTR(PACK(254,TI),60)
0051 CALL ENABLF(0)
0052 CALL ENQUE(PXINQ,XINQ,TI)
0053 CALL ENABLE(1)

```

PAGE 002

```
FORTRAN IV      V02.1-11
0054             LLFLG=0
0055             RETURN
0056             END
```


PAGE 001

EPCSTRAN IV V02.1-11

```

0001 SUBROUTINE FIAC
0002 LOGICAL*1 PRED, IDATA
0003 REAL*4 FMEAS
0004 REAL*8 FTR
0005 COMMON /FCOM/ ICONN(3,6), ILATA(50), ICHAN, MON, FIR, FMEAS,
      & ITYPE, IAREA, ITERM, PRED(1014)
0006 DECODE(2,4, ILATA(15), ERR=250) ICHAN
0007 DECODE(4,6, ILATA(19), ERR=250) FMEAS
0008 DECODE(2,4, IDATA(22), ERR=250) MON
0009 4 FORMAT(A2)
0010 6 FORMAT(A4)
0011 ITERM=IDATA(24)
0012 10 IF(IDATA(6) .EQ. 22) GOTO 20
0014 IF(IDATA(6) .EQ. 23) GOTO 30
0016 IF(IDATA(6) .EQ. 26) GOTO 40
0018 IF(IDATA(6) .EQ. 28) GOTO 50
0020 RETURN
0021 ITYPE=1
0022 GOTO 60
0023 30 ITYPE=3
0024 GOTO 60
0025 40 ITYPE=5
0026 IF((IDATA(7) .EQ. 'X') .AND.
      & (IDATA(8) .EQ. 'X') .AND.
      & (IDATA(9) .EQ. 'X') .AND.
      & (IDATA(10) .EQ. 'X')) GOTO 45
      GOTO 60
0028 45 CALL SNDRT(3)
0029 RETURN
0030 ITYPE=IDATA(25)
0031 IF(ITYPE .EQ. 1 .OR. ITYPE .EQ. 3 .OR.
0032 & ITYPE .EQ. 5) GOTO 60
      RETURN
0034 IF(IDATA(17) .EQ. 1) GOTO 70
0035 IF(ITYPE .EQ. 5) ICHAN=ICHAN+1000
0037 CALL SNDRT(2)
0039 RETURN
0040 IF(ITYPE .EQ. 5) ICHAN=ICHAN+1000
0041 PRED(ICHAN) = PRED(ICHAN)+1
0043 IF(PRED(ICHAN) .EQ. 1) GOTO 80
0046 IF(PRED(ICHAN) .GE. 21) PRED(ICHAN)=0
0048 RETURN
0049 80 CALL SNDRT(1)
0050 IF(ITYPE .EQ. 5) ICHAN=ICHAN-1000
0052 IF((MON .GE. 1) .AND.
      & (MON .LE. 333)) GOTO 90
0054 IF((MON .GE. 334) .AND.
      & (MON .LE. 666)) GOTO 100
0056 IF((MON .GE. 667) .AND.
      & (MON .LE. 1000)) GOTO 110
      RETURN
0058 90 IAREA=1
0059 GOTO 130
0060 100 IAREA=2

```

!SEND ALARM REPORT

!SEND AMBER REPORT

!SEND RED REPORT

PAGE 002

```

FORTRAN IV      V02.1-11
0062      CALL SNEPT(4)
0063      GOTO 130
0064      IAREA=3
0065      IF(ICONN(IAREA,ITYPE).EQ.2) GOTO 140
0066      RETURN
0067      ICONN(IAREA,ITYPE)=ICHAN
0068      ICONN(IAREA,ITYPE+1)=MON
0069      DO 150 I=1,6
0070      IF(ICONN(IAREA,I).EQ.2) GOTO 250
0071      CONTINUE
0072      IF(ICONN(IAREA,2).LT.ICONN(IAREA,4)) GOTO 190
0073      IF(ICONN(IAREA,4).LT.ICONN(IAREA,6)) GOTO 200
0074      IF(ICONN(IAREA,2).LT.ICONN(IAREA,6)) GOTO 210
0075      IT=5
0076      GOTO 220
0077      IT=4
0078      GOTO 220
0079      IT=2
0080      IF((IAREA.EQ.1).OR.(IAREA.EQ.2)) GOTO 230
0081      CALL FAULT(2,IT)
0082      GOTO 240
0083      CALL FAULT(1,IT)
0084      DO 245 I=1,6
0085      ICONN(IAREA,I)=0
0086      CONTINUE
0087      RETURN
0088      END

```

```

FORTRAN IV      V02.1-11      PAGE 001

0001 SUBROUTINE SDRPT(IC)
0002 REAL*8 HEAD(3),PTR
0003 INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ
0004 INTEGER*2 FLWNT,OUTFCT,OUTQ,RESNM
0005 LOGICAL*1 PACK,DAEH(24),IDATA,PRED,ISLID,MONITOR
0006 LOGICAL*1 LF,CR,ETX,DUM
0007 REAL*4 FMEAS
0008 EQUIVALENCE (HEAD,DAEH)
0009 DATA HEAD(1),TRUNK //,HEAD(2)//CHANNEL //,HEAD(3)//COND
0010 COMMON PACK(256,64)
0011 & ITYPE,IAREA,ITERM,PRED(1014)
0012 COMMON /MESS/ MESSEQ
0013 COMMON /GLOB/ ISENT,FLWNT,IQINTH,OUTFCT,IFULL,OUTQ,
0014 & RESNM,IALTHT,ISLID,MONITOR,LF,ETX,CR,DUM
0015 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0016 & ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0017 IF(IC.EQ.3) GOTO 100
0018 IF(IC.EQ.4) GOTO 200
0019 CALL ENABLE(0)
0020 K2=IGETSP(N)
0021 CALL ENABLE(1)
0022 IF(MESSEQ.EQ.126) MESSEQ=0
0023 MESSEQ=MESSEQ+1
0024 PACK(1,K2)=0
0025 PACK(2,K2)=MESSEQ
0026 PACK(3,K2)=0
0027 PACK(4,K2)=0
0028 PACK(5,K2)=24
0029 PACK(6,K2)=ISLID
0030 PACK(7,K2)=177
0031 DO 70 I=7,14
0032   PACK(I+1,K2)=IDATA(I)
0033 CONTINUE
0034 ENCODE(2,80,PACK(16,K2)) ICHAN
0035 FORMAT(A2)
0036 PACK(18,K2)=IC
0037 DO 86 J=19,40
0038   PACK(J,K2)=0
0039   PACK(40,K2)=CR
0040   PACK(41,K2)=LF
0041   PACK(42,K2)=ETX
0042 CALL ENSTR(PACK(254,K2),42)
0043 CALL ENABLE(0)
0044 CALL ENQUE(PXINQ,XINQ,K2)
0045 CALL ENQUE(PXOUTQ,XOUTQ,K2)
0046 CALL ENABLE(1)
0047 CONTINUE
0048 CALL ENABLE(0)
0049 K1=IGETSP(N)
0050 CALL ENABLE(1)
0051 DO 10 J=1,63
0052   PACK(J,K1)=040
0053 IF(MESSEQ.EQ.126) MESSEQ=0
0054 MESSEQ=MESSEQ+1

```

PAGE 202

```

FORTRAN IV      V02.1-11
0056      PACK(1,K1)=0
0057      PACK(2,K1)=MESSEQ
0058      PACK(3,K1)=0
0059      PACK(4,K1)=0
0060      PACK(5,K1)=ITERM
0061      PACK(6,K1)=ISLID
0062      DO 20 I=1,5
0063      PACK(I+9,K1)=DAEH(I)
0064      CONTINUE
0065      DO 30 I=9,15
0066      PACK(I+11,K1)=DAEH(I)
0067      CONTINUE
0068      DO 40 I=17,20
0069      PACK(I+14,K1)=DAEH(I)
0070      CONTINUE
0071      PACK(35,K1)=CR
0072      PACK(36,K1)=LF
0073      DO 50 I=7,14
0074      PACK(I+32,K1)=IDATA(I)
0075      CONTINUE
0076      ENCODE(4,60,PACK(51,K1)) ICHAN
0077      FORMAT(14)
0078      ENCODE(2,65,PACK(61,K1)) IC
0079      FORMAT(12)
0080      PACK(63,K1)=CR
0081      PACK(64,K1)=LF
0082      PACK(65,K1)=ETX
0083      CALL ENSTR(PACK(254,K1),65)
0084      CALL ENABLE(0)
0085      CALL ENQUE(PXINQ,XINQ,K1)
0086      CALL ENABLE(1)
0087      RETURN
0088      CALL ENABLE(0)
0089      K1=IGETSP(N)
0090      K2=IGETSP(N)
0091      CALL ENABLE(1)
0092      DO 110 I=1,256
0093      PACK(I,K1)=40
0094      PACK(I,K1)=40
0095      CONTINUE
0096      IF(MESSEQ.EQ.126) MESSEQ=0
0097      MESSEQ=MESSEQ+1
0098      PACK(1,K1)=0
0099      PACK(2,K1)=MESSEQ
0100      PACK(3,K1)=0
0101      PACK(4,K1)=0
0102      PACK(5,K1)=ITERM
0103      PACK(6,K1)=ISLID
0104      DO 120 I=7,10
0105      PACK(I,K1)=007
0106      CONTINUE
0107      DO 130 I=7,10
0108      PACK(I+4,K1)=IDATA(I)
0109      CONTINUE
0110

```

PAGE 003

```

FORTRAN IV      V02.1-11

0111  ENCODE(2,140,PACK(16,K1)) IDATA(11)
0112  ENCODE(2,140,PACK(18,K1)) IDATA(13)
0113  FORMAT(I2)
0114  PACK(20,K1)=CR
0115  PACK(21,K1)=LF
0116  PACK(22,K1)=TX
0117  CALL ENSTR(PACK(254,K1),22)
0118  IF(MESSEQ.EQ.126) MESSEQ=0
0119  MESSEQ=MESSEQ+1
0120  PACK(1,K2)=0
0121  PACK(2,K2)=MESSEQ
0122  PACK(3,K2)=0
0123  PACK(4,K2)=0
0124  PACK(5,K2)=24
0125  PACK(6,K2)=ISLID
0126  DO 150 I=7,14
0127     PACK(I,K2)=IDATA(I)
0128  CONTINUE
0129  150
0130  PACK(18,K2)=IC
0131  PACK(19,K2)=015
0132  PACK(20,K2)=012
0133  PACK(21,K2)=003
0134  CALL ENSTR(PACK(254,K2),21)
0135  CALL ENABLE(0)
0136  CALL ENQUE(PXING,XING,K1)
0137  CALL ENQUE(PXING,XING,K2)
0138  CALL ENABLE(1)
0139  RETURN
0140  200
0141  CALL ENABLE(0)
0142  K1=ICETSP(N)
0143  CALL ENABLE(1)
0144  IF(MESSEQ.EQ.126) MESSEQ=0
0145  MESSEQ=MESSEQ+1
0146  PACK(1,K1)=0
0147  PACK(2,K1)=MESSEQ
0148  PACK(3,K1)=0
0149  PACK(4,K1)=0
0150  PACK(5,K1)=28
0151  PACK(6,K1)=ISLID
0152  PACK(7,K1)=5
0153  DO 210 I=7,14
0154     PACK(I-1,K1)=IDATA(I)
0155  CONTINUE
0156  ENCODE(2,220,PACK(16,K1)) ICHAN
0157  220
0158  FORMAT(A2)
0159  PACK(18,K1)=IC
0160  CALL ENSTR(PACK(254,K1),18)
0161  CALL ENABLE(0)
0162  CALL ENQUE(PXING,XING,K1)
0163  CALL ENABLE(1)
0164  RETURN
0165  END

```



```

FORTRAN IV      V02.1-11      PAGE 001

0001 SUBROUTINE FAULT(IC,IT)
0002 LOGICAL*1 MSG1(40),MSG2(56),PACK,IData,PRED
0003 LOGICAL*1 CR,CTX,DUM,ISLID,LF,MONTOR
0004 INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ
0005 INTEGER*2 FLWNT,OUTFCT,OUTQ,RESNLM
0006 REAL*4 FMEAS
0007 REAL*8 MSG1(5),MSG2(7),PTR
0008 EQUIVALENCE (MSG1,IMSG1),(MSG2,IMSG2)
0009 COMMON PACK(256,64)
0010 COMMON /MESS/ MESSEQ
0011 COMMON /FCOM/ ICONN(3,6),IDATA(80),ICHAN,MON,PTR,FMEAS,
0012 & ITYPE,IAREA,ITERM,PRED(1014)
0013 & RESNLM,IALTRT,ISLID,MONTOR,LF,CTX,CR,DUM
0014 & ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0015 & DATA MSG1(1),'FAULT AR',MSG1(2),'EA CAUS',MSG1(3),'ED BY ',
0016 & MSG1(4),'LINECHAN',MSG1(5),'MON PT ',
0017 & MSG2(1),'FAULT CA',MSG2(2),'NNOT BE ',MSG2(3),'ISOLATED',
0018 & MSG2(4),'POSSIBLE',MSG2(5),'CAUSE ',MSG2(6),'LINECHAN',
0019 & MSG2(7),'MON PT '
0020 CALL ENABLE(0)
0021 K1=IGETSP(N)
0022 K2=IGETSP(N)
0023 CALL ENABLE(1)
0024 IF(IC -EQ. 2) GOTO 100
0025 DO 80 ICRT=18,25,7
0026 K3=K1
0027 IF(ICRT -EQ. 25) K3=K2
0028 IF(MESSEQ -EQ. 126) MESSEQ=0
0029 MESSEQ=MESSEQ+1
0030 PACK(1,K3)=0
0031 PACK(2,K3)=MESSEQ
0032 PACK(3,K3)=0
0033 PACK(4,K3)=0
0034 PACK(5,K3)=ICRT
0035 PACK(6,K3)=ISLID
0036 DO 10 I=1,8
0037 PACK(I+6,K3)=IMSG1(I)
0038 DO 20 I=9,11
0039 PACK(I+6,K3)=IMSG1(I)
0040 CONTINUE
0041 ENCODE(2,25,PACK(18,K3)) IC
0042 FORMAT(I2)
0043 DO 30 I=12,16
0044 PACK(I+8,K3)=IMSG1(I)
0045 CONTINUE
0046 DO 35 I=17,22
0047 PACK(I+8,K3)=IMSG1(I)
0048 CONTINUE
0049 IF(IT -EQ. 6) GOTO 45
0050 DO 40 I=29,32
0051 PACK(I+2,K3)=IMSG1(I)
0052

```

PAGE 002

```

FORTRAN IV      V02.1-11
0053 40 CONTINUE
0054 GOTO 55
0055 45 DO 50 I=25,28
0056      PACK(I+6,K3)=MSG1(I)
0057 50 CONTINUE
0058 55 ENCODE(5,60,PACK(35,K3)) ICONN(IAREA,IT-1)
0059 50 FORMAT(I5)
0060 DO 70 I=33,40
0061      PACK(I+7,K3)=MSG1(I)
0062 70 CONTINUE
0063 ENCODE(5,60,PACK(48,K3)) ICONN(IAREA,IT)
0064      PACK(53,K3)=CR
0065      PACK(54,K3)=LF
0066      PACK(55,K3)=ETX
0067 CALL ENSTR(PACK(254,K3),55)
0068 CONTINUE
0069 GOTO 210
0070 100 DO 200 ICRT=18,25,7
0071      K3=K1
0072      IF(ICRT.EQ.25) K3=K2
0073      IF(MESSEQ.EQ.126) MESSEQ=0
0074      MESSEQ=MESSEQ+1
0075 200 CONTINUE
0076      PACK(1,K3)=0
0077      PACK(2,K3)=MESSEQ
0078      PACK(3,K3)=0
0079      PACK(4,K3)=0
0080      PACK(5,K3)=ICRT
0081      PACK(6,K3)=ISLID
0082 DO 110 I=1,8
0083      PACK(I+6,K3)=MSG2(I)
0084 110 CONTINUE
0085 DO 120 I=9,16
0086      PACK(I+6,K3)=MSG2(I)
0087 120 CONTINUE
0088 DO 130 I=17,24
0089      PACK(I+6,K3)=MSG2(I)
0090 130 CONTINUE
0091      PACK(31,K3)=CR
0092      PACK(32,K3)=LF
0093 DO 140 I=25,32
0094      PACK(I+8,K3)=MSG2(I)
0095 140 CONTINUE
0096 DO 150 I=33,40
0097      PACK(I+8,K3)=MSG2(I)
0098 150 CONTINUE
0099      IF(IT.EQ.6) GOTO 165
0100 DO 160 I=41,44
0101      PACK(I+8,K3)=MSG2(I)
0102 160 CONTINUE
0103 GOTO 175
0104 DO 170 I=45,48
0105      PACK(I+4,K3)=MSG2(I)
0106 170 CONTINUE
0107 ENCODE(5,180,PACK(53,K3)) ICONN(IAREA,IT-1)
0108 175
0109

```

PAGE 003

```

FORTRAN IV      V22.1-11
0110 180      FORMAT(15)
0111          DO 190 I=49,56
0112          PACK(I+9,K3)=MSG2(1)
0113 190      CONTINUE
0114          ENCODE(5,180,PACK(56,K3)) ICONN(IAREA,IT)
0115          PACK(71,K3)=CR
0116          PACK(72,K3)=LF
0117          PACK(73,K3)=17
0118          PACK(74,K3)=ETX
0119          CALL ENSTR(PACK(254,K3),74)
0120 200      CONTINUE
0121 210      CALL ENABLE(0)
0122          CALL ENQUE(PXING,XING,K1)
0123          CALL ENQUE(PXING,XING,K2)
0124          CALL ENABLE(1)
0125          RETURN
0126          END

```

FDM,MACRO

MACRO V03.02B 00:36:52 PAGE 1

```

1  .TITLE FDM,MACRO
2  .SETTL MODE 27
3  .ILENT /V3.0/
4  .GLOBL LIUMT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WOKEN,RAM,STATUS,STATES
6  .GLOBL IPOUT,RSSTART,RPNT,DETR,ENSTR,LPIAPT
7  .NLIST CND
8  .PSECT
9
10 000000
11
12 000001
13 000002
14 000003
15 000004
16 000005
17 000006
18 000007
19
20
21 172410
22 172412
23 172414
24 172416
25 172416
26 172416
27 177560
28 177562
29 177564
30 177564
31 177565
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65

```

; CHANGED

```

; INTERFACE ADDRESS
BAR= 172410
WCR= 172412
CSR= 172414
IOBUF= 172416
OPREQ= 172416
HRCSP= 177560
HRBUF= 177562
HXCSR= 177564
HXBUF= 177565

; INTERFACE OPTIONS
XCIO=1
XSIO=1
XGIO=1
NDXX=0
ND24=1
NEOST=0
DHOST=1

; LOCAL VARIABLES
ARG1= 2
ARG2= 4
ARG3= 6
ARG4= 10
BIT5= 100000
BIT14= 40000
BIT08= 400
BIT07= 200
BIT06= 100
BIT05= 40
BIT04= 20
BIT03= 10
BIT02= 4

```

66 67 68

000002
000001

BIT01 = 2
BIT00 = 1


```
1 0000000 012700 000340
2 0000004 105400
3 0000006 012700 000124
4 0000012 012720 000412
5 0000016 012710 000340
6 0000022 012700 000100
7 0000026 012710 001636
8 0000032 042737 040100
9 0000036 004767 000014
10 0000040 005037 172414
11 0000044 012700 000000
12 0000050 105400
13 0000054 002027
14 0000056
15 0000060
16 0000064
17 0000068
18 0000072
19 0000076
20 0000080
21 0000084
22 0000088
23 0000092
24 0000096
25 0000100
26 0000104
27 0000108
28 0000112
29 0000116
30 0000120
31 0000124
32 0000128
33 0000132
34 0000136
35 0000140
36 0000144
37 0000148
38 0000152
39 0000156
40 0000160
41 0000164
42 0000168
43 0000172
44 0000176
45 0000180
46 0000184
47 0000188
48 0000192
49 0000196
50 0000200
51 0000204
52 0000208
53 0000212
54 0000216
55 0000220
56 0000224
57 0000228
58 0000232
59 0000236
60 0000240
61 0000244
62 0000248
63 0000252
64 0000256
65 0000260
66 0000264
67 0000268
68 0000272
69 0000276
70 0000280
71 0000284
72 0000288
73 0000292
74 0000296
75 0000300
76 0000304
77 0000308
78 0000312
79 0000316
80 0000320
81 0000324
82 0000328
83 0000332
84 0000336
85 0000340
86 0000344
87 0000348
88 0000352
89 0000356
90 0000360
91 0000364
92 0000368
93 0000372
94 0000376
95 0000380
96 0000384
97 0000388
98 0000392
99 0000396
100 0000400
101 0000404
102 0000408
103 0000412
104 0000416
105 0000420
```

```
*****
***** PROCEDURE MASTER START UP *****
*****
*****
MASTER: MOV #340, R0
          MGV R0
          MOV #124, R0
          MOV #L10, (R0)+
          MOV #340, (R0)
          MOV #100, R0
          MOV #TIME, (R0)
          BIC #40100, @CSR
          JSR PC, LIUINT
          CLR @CSR
          MOV #000, R0
          MTPS R0
          RTS PC
          PC

          ;LIU HANDLER
          ;PRI=7
          ;CLOCK VECTOR
          ;CLOCK
          ;DISABLE INTERRUPTS
          ;CLEAR LIU
          ;CLEAR BLUI CSR
          ;PRI=0
          ;CPU DOWN

          ;LDADR
          ;ADDRESS=0
          ;GOOD WD
          ;NO RETRY
          ;SEL ACRAM
          ;COUNTER
          ;WRITE A NULL
          ;GOOD WD
          ;NO LOOP UNTIL READY
          ;COUNT+1

          ;RDRUFADR CMD
          ;SEL INBUFO
          ;SET POINTER=0
          ;FALSE READ DATA
          ;GOOD READ
          ;NO RETRY
          ;CLEAR DONE BIT
          ;RDRUFADR COMMAND
          ;SEL INBUF1 COMMAND
          ;POINTER=0
          ;FALSE READ DATA
          ;GOOD RD
          ;NO RETRY
          ;CLEAR DONE BIT
```

```
*****
***** PROCEDURE INITIALIZE *****
*****
*****
          ;CLEAR ACRAM
          LIUINT: MOV #4354., @OPREG
                  MOV #2304., @OPREG
                  TSTB @CSR
                  BPL -4
                  CLRB @CSR
                  MOV #4353., @OPREG
                  MOV #-256., R0
                  MOV #2311., @OPREG
                  TSTB @CSR
                  BPL -4
                  CLRB @CSR
                  INC R0
                  BNE 1$
          ;CLEAR INPUT/OUTPUT BUFFERS
          MOV #10410, R1
          MOV #10440, R0
          JSR PC, ZEROBP
          MOV #1400, @OPREG
          TSTB @CSR
          BPL -4
          CLRB @CSR
          MOV #10510, R1
          MOV #10640, R0
          JSR PC, ZEROBP
          MOV #1420, @OPREG
          TSTB @CSR
          BPL -4
          CLRB @CSR
```



```

1 ***** PROCEDURE LIU-HANDLER *****
2 ;
3 ;
4 ;
5 ;
6 LIO:      MOV      R0,      -(SP)
7           MOV      R1,      -(SP)
8           MOV      R2,      -(SP)
9           MOV      R3,      -(SP)
10          MOV      R4,      -(SP)
11          MOV      R5,      -(SP)
12          TST      @RCSR
13          BPL      R15
14          BIC      #BIT14, @RCSR
15          MOV      #352., @OPREG
16          MOV      #1280., @OPREG
17          MOV      @R0BUF, CAUSE
18          BIC      #172400, CAUSE
19
20 ;PERFORM OPERATION DEPENDING ON BITS
21
22          BITB     #BIT02, CAUSE
23          BEQ      R3$,
24              MOV      #4360., R2
25              MOV      #4384., R3
26              JSR      PC, EMBF
27              BITB     #BIT03, CAUSE
28              BEQ      R4$,
29                  MOV      #4488., R2
30                  MOV      #4512., R3
31                  JSR      PC, EMBF
32              BITB     #BIT04, CAUSE
33              BEQ      R5$,
34                  MOV      #BIT07, CAUSE
35                  BEQ      R6$,
36                      MOV      #2, LLFG
37                      BITB     #BIT05, CAUSE
38                      BEQ      R7$,
39                          MOV      #1, LLFG
40                          BITB     #BIT06, CAUSE
41                          BEQ      R8$,
42                              CLR      IWRITM
43                              CLR      IWRIT
44          ;BUFFER FULL
45
46          BITB     #BIT00, CAUSE
47          BEQ      R9$,
48              MOV      #INBF, R1
49              MOV      #360., R2
50              MOV      #384., R3
51              MOV      #BIT00, R4
52              JSR      PC, EMBF
53              MOV      #AREA, R5
54              MOV      #1, AREA
55              MOV      #2, LATA
56              JSR      PC, LPINPT
57

```

58 000704	132767	000002	000254	BITB	#BIT01, CAUSE	
59 000712	014423	000400		BEQ	RTI\$	
60 000714	012701	000400		MOV	#1EFF, R1	
61 000720	012702	010610		MOV	#4488, R2	
62 000724	012703	010642		MOV	#4512, R3	
63 000730	012704	000002		MOV	#BIT01, R4	
64 000734	004767	000046		JSR	PC, EMBF	
65 000740	012705	001170		MOV	#AREA, R5	
66 000744	012707	000001	000216	MOV	#1, AREA	
67 000752	012757	000216		MOV	R2, DATA	
68 000756	004767	000000G		JSR	PC, LFINPT	
69						
70 000762	012605			MOV	(SP)+, R5	
71 000764	012604			MOV	(SP)+, R4	
72 000766	012603			MOV	(SP)+, R3	
73 000770	012602			MOV	(SP)+, R2	
74 000772	012601			MOV	(SP)+, R1	
75 000774	012600			MOV	(SP)+, R0	
76 000776	052737	040000	172414	BIS	#BIT14, G#CSR	
77 001004	000002			RTI		
78						
79 001006	012737	010600	172416	MOV	#4480, G#OPREG	
80 001014	012737	002400	172416	MOV	#1280, G#OPREG	
81 001022	013700	172416		MOV	G#IOBUF, R0	
82 001026	130400			BITB	R4, R0	
83 001030	001002			BNE	CRCOK	
84 001032	012704	177777		MOV	#-1, R4	
85 001036	012737	172416		MOV	R2, G#OPREG	
86 001042	012737	001400	172416	MOV	#768, G#OPREG	
87 001050	105737	172414		TSTB	G#CSR	
88 001054	100375			BPL	-4	
89 001056	013702	172416		MOV	G#IOBUF, R2	
90 001062	042702	177400		BIC	#177400, R2	
91 001066	010200			MOV	R2, R0	
92 001070	005400			NEG	R0	
93 001072	010237	172412		MOV	R0, G#CR	
94 001076	010137	172410		MOV	R1, G#EAR	
95 001102	010337	172416		MOV	R3, G#OPREG	
96 001106	012737	001400	172416	MOV	#768, G#OPREG	
97 001114	105737	172414		TSTB	G#CSR	
98 001120	100375			BPL	-4	
99 001122	012737	021000	172416	MOV	#2704, G#OPREG	
100 001130	000240			NOP		
101 001132	105737	172414		TSTB	G#CSR	
102 001136	100401			BMI	DMAOK	
103 001140	000240			NOP		
104 001142	012737	004400	172416	MOV	#2304, G#OPREG	
105 001150	105737	172414		TSTB	G#CSR	
106 001154	100375			BPL	-4	
107 001156	005704			TST	R4	
108 001160	100001			BPL	END\$	
109 001162	010402			MOV	R4, R2	
110 001164	000207			RTS	PC	
111 001166	000000			CAUSE:	.WORD 0	
112 001170	000000			AREA:	.WORD 0	
113 001172	001174			DATA:	.WORD 0	
114 001174	000000					

```

;INBUF1 FULL
;NO MORE DONE
;BUFFER ADDRESS
;RDBUFADR COMM
;SEL INBUF1
;CRC BIT
;GO EMPTY BUFFER
;DATA LINK AREA
;ONE VARIABLE
;CRC OR BYTE COUNT
;CALL FORTRAN QUE'ER
;RESTORE REGISTERS

;ENABLE INTERRUPTS
;RETURN FROM INTERRUPT

;READ STATUS 1
;RS
;FETCH STATUS
;GOOD CRC

;NO FLAG
;RDBUFADR
;RD
;GOOD RD

;REPLACE WITH POINTER
;CLEAR MST BITS
;SAVE IT
;Z'S COMP
;BYTE COUNT NOW
;ADDRESS IN MEMORY
;SEL BUFFER
;FALSE RD
;DONE ON
;NO LOOP
;FIRE DMA
;DELAY
;GOOD DMA

;ERROR IF HERE
;FALSE WD
;GOOD WD

;WAS CRC OK
;YES
;NO FLAG IT
;RETURN
;STATUS BYTE 0 HOLDER

```



```

1  10 001306 017501 000002 172416 010400 000001 000001
2  11 001312 022701 000001 172416 010400 000001 000001
3  12 001316 001415 000001 172416 010400 000001 000001
4  13 001320 012737 000001 172416 010400 000001 000001
5  14 001326 012737 000001 172416 010400 000001 000001
6  15 001334 013700 000001 172416 010400 000001 000001
7  16 001340 042700 000001 172416 010400 000001 000001
8  17 001344 010075 000001 172416 010400 000001 000001
9  18 001350 000207 000001 172416 010400 000001 000001
10 19 001352 012737 000001 172416 010400 000001 000001
11 20 001360 012737 000001 172416 010400 000001 000001
12 21 001366 013700 000001 172416 010400 000001 000001
13 22 001372 042700 000001 172416 010400 000001 000001
14 23 001376 010075 000001 172416 010400 000001 000001
15 24 001402 000207 000001 172416 010400 000001 000001
16 25 001404 012737 000001 172416 010400 000001 000001
17 26 001412 012737 000001 172416 010400 000001 000001
18 27 001420 013701 000001 172416 010400 000001 000001
19 28 001424 042701 000001 172416 010400 000001 000001
20 29 001430 132701 000001 172416 010400 000001 000001
21 30 001434 001404 000001 172416 010400 000001 000001
22 31 001436 012775 000001 172416 010400 000001 000001
23 32 001444 000207 000001 172416 010400 000001 000001
24 33 001446 012775 000001 172416 010400 000001 000001
25 34 001454 000207 000001 172416 010400 000001 000001
26 35
27 36

*****
***** PROCEDURE STATUS *****
*****
;CALL STATUS(X,DATA) -READ STATUS BYTES 0/1
;
;X=0 STATUS BYTE 0
;X=1 STATUS BYTE 1
;
;WHICH BYTE
;COMPARE
;WCR : RS(0)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;WCR : RS(1)
;RS
;FETCH DATA
;CLEAR BITS
;RTN DATA
;STATUS BYTE 1
;RS
;FETCH DATA
;BUFFER FULL
;LOAD RETURN
;LOAD RETURN

```


FM-MACRO MACRO V03.22B 20:36:52 PAGE 7

FM-MACRO
NOTE 27

```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
*****
***** PROCEDURE ACRAM *****
*****
;CALL RAM(2,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;2100 =DREAD
;0111 =NULL
;0000 =WOKEN

RAM: MOV @ARG1(R5), R0
      CMP #0, R0
      BNE RDRAM
      WTRAM: MOV @ARG2(R5), R0
            MOV @ARG3(R5), R1
            MOV #4354., @OPREG
            ADD #2304., R0
            MOV R0, @OPREG
            TSTB @CSCR
            BPL -4
            CLRB @CSCR
            MOV #4353., @OPREG
            ADD #2304., R1
            MOV R1, @OPREG
            TSTB @CSCR
            BPL -4
            CLRB @CSCR
            RTS
      RDRAM: MOV @ARG2(R5), R0
            MOV #4354., @OPREG
            ADD #2304., R0
            MOV R0, @OPREG
            TSTB @CSCR
            BPL -4
            CLRB @CSCR
            MOV #4353., @OPREG
            MOV #768., @OPREG
            TSTB @CSCR
            BPL -4
            CLRB @CSCR
            MOV @IOBUF, R0
            BIC #177760, R0
            MOV R0, @ARG3(R5)
            RTS

; RAM
; WHICH OPERATION
; READ OF
; ADDRESS
; WRITE DATA
; SEL LDACR
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; WD/DATA (CMD)
; WRITE
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; ADDRESS TO READ
; SEL LDACR
; WD/DATA (ADDR)
; WRITE DATA
; VALID WRITE
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; SEL ACRAM
; READ DATA
; VALID READ
; NO LOOP UNTIL READY
; CLEAR DONE BIT
; FETCH DATA
; CLEAR BITS
; RTN DATA

```

```

1 1 *****.ENABLE LSB*****
2 2 *****PROCEDURE WRITE LOOP (LPOUT)*****
3 3 *****
4 4 *****
5 5 *****
6 6 002064 017502 000002 LPOUT: MOV GARG1(R5), R2 ; BYTE COUNT
7 7 002070 012700 000340 MOV #340, R0 ; PRI=7
8 8 002074 106400 MTPS R0, R0 ; BUS ADDRESS
9 9 002076 012737 000000' 172410 MOV #OUTEF, G#BAR ; 2'S COMP COUNT
10 10 002104 005402 NEG R2 R2 ; COUNT
11 11 002106 012337 172412 MOV R2, G#WCR ; DMA COMMAND
12 12 002112 012737 010540 MOV #448., G#OPREG ; DMA GO
13 13 002120 012737 024000 172416 MOV #10240., G#OPREG ; INTER ACE TIME
14 14 002126 000240 NOP ; DMA OK
15 15 002130 105737 172414 TSTB G#CSR ; ERROR IF HERE
16 16 002134 000240 NOP ; OB1 COMMAND
17 17 002136 105037 172414 CLRB G#CSR ; ADDRESS OF WRITE TOKEN
18 18 002142 012737 010740 172416 MOV #4576., G#OPREG ; BYTE COUNT
19 19 002150 012737 001634' 172410 MOV #1D, G#BAR ; FOR AT FOR BLUI
20 20 002156 012702 000002 NEG R2 R2 ; DMA BYTE COUNT
21 21 002162 005402 MOV R2, G#WCR ; FIRE DMA
22 22 002164 012737 172412 MOV #10240., G#OPREG ; DELAY
23 23 002170 012737 024000 172416 TSTB G#CSR ; GOOD DMA
24 24 002176 000240 NOP ; ERROR IF NOT
25 25 002200 105737 172414 CLRB G#CSR ; CLEAR DONE
26 26 002204 000240 MOV #4368., G#OPREG ; MODSTAT
27 27 002206 105037 172414 MOV #2307., G#OPREG ; BUFFERS FULL
28 28 002212 012737 010420 TSTB G#CSR ; GOOD WRITE
29 29 002220 012737 004403 BPL -4 ; NO LOOP UNTIL
30 30 002226 105737 172414 CLRB G#CSR ; CLEAR DONE BIT
31 31 002232 100375 172414 MOV #000, R0 ; PRI=0
32 32 002234 105037 000000 MTPS R0, PC ; LOWER CPU
33 33 002240 012700
34 34 002244 106400
35 35 002246
36 36
37 37
38 38

```


EDM-MACRO
NODE 27

MACRO V03.02B 00:36:52 PAGE 10

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 002376 017500 000002
10 002402 022700 000000
11 002406 001002
12 002410 000000
13 002412 000207
14 002414 022700
15 002420 001005
16 002422 004767 175432
17 002426 012700 000040
18 002432 000110
19 002434 022700 000002
20 002440 001002
21 002442 000177 000002
22 002446 000207
23 002450 173000
24
25
26
243

```

```

*****
***** PROCEDURE RESTART *****
*****
*****
;CALL RSTART(0) -CAUSES A SOFTWARE HALT
;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
;CALL RSTART(2) -LOAD MODE(173000)

RSTART: MOV  GARG1(R5), R0
RS0:  CMP  #0, R0
      BNE  RS1
      HALT
      RTS  PC
RS1:  CMP  #1, R0
      BNE  RS2
      JSR  PC, LIUINT
      MOV  #40, R0
      JMP  (R0)
RS2:  CMP  #2, R0
      BNE  RST
      JMP  GROM
RST:  RTS  PC
ROM:  .WORD 173000

;MODE
;LOAD ADDRESS

;RESTART PROGRAM
;INT LIU FIRST
;START ADDRESS
;LOAD MODE

```

EDM MACRO
NOTE 27

MACRO V03.02B 00:36:52 PAGE 11

```

1
2
3
4
5 000000
6
7 000000
8 000400
9 001000
10 001002
11 001004
12 001006
13
14
15 000000
16
17 000000
18 000400
19 001000
20 001400
21 002000
22 002002
23 002004
24
25
26
27

*****
***** COMMON DATA AREA *****
*****
.PSECT DFM,RW,D,GBL,REL,OVR

OUTBF: .BLKB 256.
INBF: .BLKB 256.
IWRITM: .BLKW
IWRIT: .BLKW
STAT: .BLKW
LTIME: .BLKW
.EVEN

.PSECT BUFS,RW,D,GBL,REL,OVR

XOUTBF: .BLKB 256.
XINBF: .BLKB 256.
ACKSEQ: .BLKB 256.
LIDFD: .BLKB
IOFLG: .BLKW
IRSEND: .BLKW
.EVEN

.END
000001

```

EDM.MACRO MACRO V03.02B 00:36:52 PAGE 11-1

SYMBOL TABLE

```

ACKSEQ 001000R
AREA 001170R
ARG1 = 000002
ARG2 = 000004
ARG3 = 000006
ARG4 = 000010
EAR = 172410
BIT00 = 000001
BIT01 = 000002
BIT02 = 000004
BIT03 = 000010
BIT04 = 000020
BIT05 = 000040
BIT06 = 000100
BIT07 = 000200
BIT08 = 000400
BIT14 = 040000
BIT15 = 100000

003 B0$ 000626R
B1$ 000704R
B2$ 000472R
B3$ 000516R
B4$ 000542R
B5$ 000570R
B6$ 000606R
B7$ 000552R
CAUSE 001166R
CROCK 001036R
CSR = 172414
DATA 001174R
DESTR 001240RG
DHOST = 000001
DMAOK 001142R
EMBP 001006R
ENABLE 001175RG
END$ 001164R

ENSTR 001230RG
E0$ 001210R
E1$ 001220R
G4$ 002426R
HRRBF = 177562
HRCSE = 177560
HXBUFF = 177566
HXCSR = 177564
INBF 000400R
IOBUF = 172416
IOFLG 002000R
IRSEND 002004R
IWRT 001002R
IWRTM 001000R
LIDFD 001400R
LIO 000412RG
LIUINT 000060RG
LLFLG 002002R

LPINPT = ***** G
LPOUT 002264RG
LTIME 001006R
MASTER 000000RG
NDAX = 000000
ND24 = 000001
NHOST = 000000
OPREG = 172416
OUTBF 000000R
RAM 001660RG
RDPM 002250RG
RDRAM 001764R
ROM 002450R
RST 002446R
RSTART 002376RG
RS0 002402R
RS1 002414R
RS2 002434R

RTI$ 000762R
STAT 001004R
STATUS 001404RG
STATUS 001306RG
STATUS 000366R
SWITCH 001250RG
TIME 001630RG
WCR = 172412
WTD 001634R
WTKG 001534R
WTKEN 001456RG
WTRAM 001672R
XCIO = 000001
XGIO = 000001
XINBF 000400R
XOUTBF 000000R
XSIO = 000001
ZEROP 000266R

```

```

. ABS. 000000 000
      002452 001
DPM 001010 002
EUPS 002006 003
ERRORS DETECTED: 0

```

VIRTUAL MEMORY USED: 300 WORDS (2 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 56 PAGES
 LK:EDM,PK:EDM=DK:EDM.M27

PAGE 001

12:00:00

01-JUL-79

CONF27.COM

PROGRAM:

DECRIPTER

FOR NOHAL/NOI LNE
FOR NOBAL1/NOI LNE
FOR FIAC/NOI LNE
MAC FOR

PAGE 001

12:00:00

01-JUL-79

LNK27.COH

PROGRAM:

DOCUMENT:

ASSIGN DXO: DN:

R LINK

DX1:NODE 27,DX1:MAP=DX1:ROBGL/C/L/W

DX1:FDR/C

DX1:ROBGL I/C

DX1:FLGL //

*SIMPL

ASSIGN DX1: DN:

1.9 NODE 28 (SDCA):

The Switch Data Collection and Analysis (SDCA) module receives switch traffic data generated by AUTODIN or AUTOVON switches, and performs loading assessments on this data to detect switch equipment saturation conditions. Traffic flow control computations and actions are performed. Simulated data to represent two switches is generated by the PDP 11/40 in loop 2. A switch condition report is sent by the PDP 11/40 to the SDCA approximately every 2.5 seconds.

SDCA Switch Condition Report

<u>Field</u>	<u>Description</u>	<u>Critical AUTODIN</u>	<u>Values AUTOVON</u>
1	Switch #	1	2
2	# of Transactions	512	256
3	# of Blocked Transactions	25	10
4	Transaction Queue Depth	25	10
5	# of Prescribed Transactions	10	25
6	Trunk Group Occupancy	50	40
7	Trunk Group Overflow	50	40
8	Message Delay (sec.)	10	5
9	Maximum Message Age (sec.)	10	5
10	Number of Overflow Messages	10	10
11	# of Senders	128	64
12	# of Markers	128	64
13	# of Receivers	128	64
14	# of Pooled Crypto Units	128	64
15	Service Time for Dial Tone (sec.)	10	5
16	Service Time for Crypto Unit (sec.)	10	5

Whenever a critical value of the table above is exceeded and the Event Reporting Condition Parameter is ON, the switch is considered to be in a saturated condition and Red Event Reports are sent to the DBMS Status File and the destination Node Designator for OCRI reporting.

The SDCA interprets commands from the DBMS similar to the VSQC, DSQC and BWBSA except that measurements are performed for 2 switches rather than channels or links.

1.9.1 Program Descriptions

1.9.1.1 Refer to Section 1.1 for descriptions of routines NODAL, IGETSP, ENQUE, DEQUE, ACKNAK, INPTQ, LPINPT, INIT, LINLOS, MASTER, LIUINT, LIO, ENABLE, SWITCH, STATUS, WTOKEN, TIME RAM, LPOUT, RDPNT, RSTART.

1.9.1.2 Subroutine SDCA (FORTRAN)

This subroutine is called when a measurement is received for the 11/40. The simulated inputs are decoded and the function described in Section 1.9 is performed.

1.9.1.3 Subroutine SDCALP (FORTRAN)

This subroutine is called when a message is received from the loop. Functions performed are reporting on or off, and request a measurement.

1.9.1.4 Subroutine IMAGE (FORTRAN)

This subroutine converts a decimal number into ASCII format.

1.9.1.5 Subroutine SIO (MACRO)

This subroutine is the interrupt handler for the 11/40 interface. It reads the message of input from the 11/40 and set a flag containing the byte count.

```

FORTRAN IV      Y02.1-11
0001      PROGRAM MODAL
0002      INTEGER*2 XING,XOUTQ,ACKQ,PXING,PXOUTQ,PACKQ
0003      INTEGER*2 PINQ,FREE,STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP
0005      INTEGER*2 RSTBKP,Q1,Q2,RESNLM,OUTQ,DEQUE
0006      LOGICAL*1 XOUTF,XINBF,ACKSEQ,LIDED,OUTF,INBF
0007      LOGICAL*1 SAVE1,SAVE2
0008      LOGICAL*1 IDATA,ENC
0009      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID,DUM
0010      INTEGER*2 LTIME,ACKTIM,NEWTIM,OLDTIM,IWRTIM,TIMLIM,ATIMLM
0011      COMMON /MESS/ MESSEQ
0012      COMMON /DFM/ OUTF(256),INBF(256),
0013      1 IWRTIM,IWRT,STAT,LTIME
0014      1 COMMON /QUE/ XING(16),PXING(2),XOUTQ(16),PXOUTQ(2),
0015      1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0016      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0017      1 LIDED(256),IOFLG,LLFLG,IRSEND
0018      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0019      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0020      COMMON /GLOB/ ISENT,FLWCNT,IQLATH,OUTFCT,IFULL,OUTQ,
0021      1 RESNLM,IALTFT,ISLID,MONTOR,LF,ETX,CR,DUM
0022      1 COMMON /SCJM/SAVE1(64),SAVE2(64),ITEM,ICOUNT,I1,I2,IDATA(256),
0023      1 ISWCH
0024      COMMON /SCM1/ISIM,IVC,IDC,IBC,ICC
0025      COMMON /END/ENC(4)
0026      CALL MASTER
0027      CALL INIT
0028      CALL ENABLE(1)
0029      5 CONTINUE
0030      25 IF(PINQ(1).LT.PINQ(2)) GOTO 40
0031      CALL ENABLE(0)
0032      Q2=DEQUE(PINQ,INQ,1)
0033      CALL ENABLE(1)
0034      DO 30 I=1,100
0035      IDATA(I)=PACK(I+6,Q2)
0036      CONTINUE
0037      CALL SDCLP
0038      IPR=IPR+1
0039      FREE(IFR)=Q2
0040      CALL STATE2(IS)
0041      IF(15.EQ.1) GOTO 100
0042      IF(IOFLG.GT.0).AND. IRSEND.EQ.0) CALL SICA
0043      IF(ILLG.GT.0) CALL LINDS
0044      IF(OUTFCT.EQ.1) GOTO 100
0045      IF((IFULL.EQ.1).OR. (ISENT.EQ.1)) GOTO 100
0046      IF(IRSEND.EQ.1) GOTO 85
0047      IF(PXING(1).LT. PXING(2)) GOTO 100
0048      CALL ENABLE(0)
0049      Q1=DEQUE(PXING,XING,1)
0050      CALL ENABLE(1)
0051      OUTQ=Q1
0052      85 CALL DESR(PACK(254,OUTQ),Q2)

```

AD-A078 391 BURROUGHS CORP PAOLI PA FEDERAL AND SPECIAL SYSTEMS GROUP F/6 9/2
SOFTWARE MAINTENANCE MANUAL FOR THE MODULAR SYSTEM CONTROL DEVE--ETC(U)
NOV 79 DCA100-76-C-0083
UNCLASSIFIED 66157 SBIE-AD-E100 313 NL

4 OF
AD
A078391



PAGE 002

```

0058      FORTRAN IV      V02.1-11
0059      DO 90 I=1,Q2
0060      OUTBF(I)=PACK(I,OUTQ)
0061      CONTINUE
0062      OUTBF(Q2+1)=0
0063      IPT=OUTBF(5)
0064      OUTBF(Q2+2)=LIDFD(IPT)
0065      IFULL=1
0066      ISENT=1
0067      CALL LPOUT(Q2+2)
0068      IRSEND=1
0069      ACKTIM=0
0070      IWRTTM=0
0071      IWRT=1
0072      INFLOT=0
0073      OLDTIM=NEWTIM
0074      NEWTIM=LTIME
0075      IF(IWRT.EQ.0) GOTO 120
0076      IWRTTM=IWRTTM+(NEWTIM-OLDTIM)
0077      IF(IWRTTM.LT.TIMLIM) GOTO 120
0078      CALL WTOKEN
0079      IWRTTM=0
0080      IWRT=0
0081      IF(ISENT.EQ.0) GOTO 130
0082      ACKTIM=ACKTIM+(NEWTIM-OLDTIM)
0083      IF(ACKTIM.LT.ATIMLIM) GOTO 130
0084      CALL ENABLE(0)
0085      CALL ACKNAK(0)
0086      CALL ENABLE(1)
0087      GOTO 5
0088      130 CONTINUE
0089      GOTO 5
0090      END
0091
0092

```

PAGE 001

```

FORTRAN IV      V02.1-11
0001      FUNCTION IGETSP(N)
0002      LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003      INTEGER*2 FREE,FL*CNT
0004      INTEGER*2 OUTTCT,OUTQ,RESNLM
0005      COMMON /FRE/ FREE(G4),IFR,IFRSZ
0006      COMMON /GLOB/ ISENT,FL*CNT,ICLNTH,OUTTCT,IFULL,OUTQ,
1          RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
C
0007      IF (IFR .LT. 1) CALL INIT
0008      IGETSP=FREE(IFR)
0009      IFR=IFR-1
0010      RETURN
0011      END
0012

```

PAGE 001

FORTRAN IV V02.1-11

```

0001 SUBROUTINE ENQUE(A,B,N)
0002 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,DUM
0003 INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWCNT,OUTQ,A(2),B(16)
0005 INTEGER*2 RESNLM,OUTFCT
0006 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0007      COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0008      RESNLM,IALTRT,ISLID,MONTOR,LF,ETX,CR,DUM
0009 IQTAIL=A(1)
0010 IF(IQTAIL.EQ.1) GOTO 20
0011 IQTAIL=IQTAIL-1
0012 B(IQTAIL)=N
0013 A(2)=IQTAIL
0014 GOTO 999
0015
0016 20 IF(IQHEAD.GE.(IQLNTH)) GOTO 40
0017 NN=IQHEAD-IQTAIL
0018 DO 30 I=1,NN+1
0019 30 B(IQLNTH+1-I)=B(IQHEAD+1-I)
0020 A(1)=IQLNTH
0021 A(2)=IQLNTH-NN
0022 GOTO 10
0023
0024 40 CALL INIT
0025 999 RETURN
0026 END

```

PAGE 001

FORTRAN IV V02.1-11

```

0001 FUNCTION DEQUE(A,B,LD)
0002 LOGICAL*1 ETX,CR,LF,DUM,ISLID,MONTOR
0003 INTEGER*2 XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ,PACKQ,INQ,PINQ
0004 INTEGER*2 FLWNT,A(2),B(16),DEQUE,OUTECT,OUTQ,RESNLM
0005 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0006      1 ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0007 COMMON /GLOB/ ISENT,FLWNT,IQLNTH,OUTECT,IFULL,OUTQ,
0008      1 RESNLM,IALTET,ISLID,MONTOR,LF,ETX,CR,DUM
0009 IQHEAD=A(1)
0010 DEQUE=B(IQHEAD)
0011 IF(ID.NE. 1) GOTO 999
0012 IF(IQHEAD.NE. 0) GOTO 10
0013 A(1)=IQLNTH
0014 A(2)=IQLNTH+1
0015 GOTO 999
0016 10 A(1)=IQHEAD-1
0017 999 RETURN
0018 END

```

PAGE 001

XOFORTRAN IV V02.1-11

```

0001 SUBROUTINE ACKNAK(N)
0002 INTEGER*2 FLWCNT,OUTFCT,STAT,XING,PXING,XOUTQ
0003 INTEGER*2 ACKQ,PACKQ,PINQ,FREQ,OUTQ,XOUTQ
0004 INTEGER*2 T1,T2,T3,T4,T5,RESNLM
0005 LOGICAL*1 ETX,CR,LF,MONITOR,ISLID,DUM
0006 LOGICAL*1 PACK,OUTF,INBF,LCOMT(44)
0007 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIIDF
0008 INTEGER*2 LTIME,IWRTTM
0009 REAL*8 COMT(5)
0010 EQUIVALENCE(COMT,LCOMT)
0011 DATA COMT(1)/'MESSAGE',/COMT(2)/'NOT SENT',/COMT(3)/' FROM ',/
0012 1 COMT(4)/'NODE',/COMT(5)/' TO NODE',/
0013 1 COMMON /GLOB/ ISENT,FLWCNT,IQLNTH,OUTFCT,IFULL,OUTQ,
0014 1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0015 1 COMMON /DFM/ OUTBF(256),INBF(256),
0016 1 IWRTTM,IWRT,STAT,LTIME
0017 1 COMMON /QUE/ PACK(256,64)
0018 1 COMMON /FREE/ FREE(64),IFR,IFRSZ
0019 1 COMMON /MESS/ MESSEQ
0020 1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),LIIDF(256),
0021 1 IOFLG,LLFLG,IRSEND
0022 IF (N.NE. 1) GOTO 10
0023 IFR=IFR+1
0024 DO 5 I=1,256
0025 PACK(I,OUTQ)=0
0026 5 CONTINUE
0027 IFULL=0
0028 ISENT=0
0029 IRSEND=0
0030 FREE(IFR)=OUTQ
0031 GOTO 999
0032 T2=PACK(256,OUTQ)
0033 IF(T2.LE.RESNLM) GOTO 120
0034 T3=PACK(3,OUTQ)
0035 T4=MOD(T3,64)
0036 IF(T4.LT.32) GOTO 110
0037 T5=IGTSP(N)
0038 IF(MESSEQ.EQ.126) MESSEQ=0
0039 MESSEQ=MESSEQ+1
0040 PACK(1,T5)=0
0041 PACK(2,T5)=MESSEQ
0042 PACK(3,T5)=0
0043 PACK(4,T5)=0
0044 PACK(5,T5)=25
0045 PACK(6,T5)=ISLID
0046 DO 20 I=7,60
0047 PACK(I,T5)=040
0048 20 CONTINUE
0049 DO 30 I=7,9
0050 PACK(I,T5)=LF
0051 30 CONTINUE

```


PAGE 002

```

FORTRAN IV      V02.1-11
0054             DO 40 I=1,8
0055             PACK(I+9,T5)=LCOMT(I)
0056             40 CONTINUE
0057             DO 50 I=9,16
0058             PACK(I+9,T5)=LCOMT(I)
0059             50 CONTINUE
0060             DO 60 I=17,22
0061             PACK(I+9,T5)=LCOMT(I)
0062             60 CONTINUE
0063             DO 70 I=25,29
0064             PACK(I+7,T5)=LCOMT(I)
0065             70 CONTINUE
0066             ENCODE(3,80,PACK(38,T5))ISLID
0067             80 FORMAT(13)
0068             DO 90 I=33,40
0069             PACK(I+11,T5)=LCOMT(I)
0070             90 CONTINUE
0071             ENCODE(3,80,PACK(52,T5))PACK(5,OUTQ)
0072             PACK(55,T5)=LF
0073             PACK(56,T5)=LF
0074             PACK(57,T5)=CR
0075             PACK(58,T5)=ETX
0076             CALL ENSTR(PACK(254,T5),58)
0077             CALL ENQUE(PXING,XING,T5)
0078             DO 102 I=1,256
0079             PACK(I,OUTQ)=0
0080             102 CONTINUE
0081             ISENT=0
0082             IREND=0
0083             IFULL=0
0084             I*RT=0
0085             IFR=IFR+1
0086             FREE(IFR)=OUTQ
0087             GOTO 999
0088             110 PACK(3,OUTQ)=PACK(3,OUTQ)+32
0089             PACK(256,OUTQ)=0
0090             ISENT=0
0091             IFULL=0
0092             IF(IREND.EQ.1) GOTO 999
0093             CALL ENQUE(PXING,XING,OUTQ)
0094             GOTO 999
0095             120 PACK(256,OUTQ)=T2
0096             IFULL=0
0097             ISENT=0
0098             IF(IASND.EQ.1) GOTO 999
0099             CALL ENQUE(PXING,XING,OUTQ)
0100             999 RETURN
0101             END
0102
0103

```

PAGE 001

```

FORTRAN IV      V02.1-11
0001      SUBROUTINE INPTQ(L)
0002      INTEGER*2 STAT,XINQ,PXINQ,XOUTQ,PXOUTQ,ACKQ
0003      INTEGER*2 PACKQ,PINQ,FLWCNT,OUTFCT,T1,RESNLM,OUTQ
0004      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFL,OUTBF,INBF
0005      LOGICAL*1 PACK,MONITOR,ISLID,ETX,CR,LF,DUM
0006      INTEGER*2 LTIME,IWRTIM
0007      COMMON /DFM/ OUTBF(256),INBF(256),
0008      1 IORTM,IWRT,STAT,LTIME
0009      1 COMMON /BUFS/ XOUTBF(256),XINBF(256),ACKSEQ(256),
0010      1 LIDFL(256),IOFLG,LLFLG,INSEND
0011      COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012      1 COMMON /GLOB/ ISENT,FLWCNT,IQNTHT,OUTFCT,IFULL,OUTQ,
0013      1 RESNLM,IALTHT,ISLID,MONITOR,LF,ETX,CR,DUM
0014      IF(L-LT.1) GOTO 30
0015      LI=L-2
0016      II=INBF(6)
0017      IF(INBF(2).EQ. ACKSEQ(II)) GOTO 20
0018      T1=IGETSP(N)
0019      DO 10 I=1,LI
0020      10 PACK(I,T1)=INBF(I)
0021      CALL ENSTH(PACK(254,T1),LI)
0022      CALL ENQUE(PINQ,INQ,T1)
0023      N=2
0024      GOTO 40
0025      20 N=1
0026      GOTO 40
0027      30 N=0
0028      40 DO 50 I=1,2
0029      50 OUTBF(I)=INBF(I)
0030      CONTINUE
0031      NN=1
0032      IF(N.EQ.0) NN=128
0033      OUTBF(3)=NN
0034      OUTBF(4)=0
0035      OUTBF(5)=INBF(6)
0036      OUTBF(6)=ISLID
0037      OUTBF(7)=ETX
0038      OUTBF(8)=0
0039      OUTBF(9)=LIDFL(INBF(6))
0040      CALL LPOUT(9)
0041      IF(N.NE.2) GOTO 999
0042      ACKSEQ(II)=INBF(2)
0043      RETURN
0044      999
0045      END
0046

```

```

FORTRAN IV      V02.1-11      PAGE 001

0001      SUBROUTINE LPINPT(LI)
0002      INTEGER*2 STAT,FLWCNT,OUTPCT,OUTC,RESNLM
0003      INTEGER*2 CC1,CC2,CC3,CC4,CC5,CC6,CC7
0004      LOGICAL*1 OUTBF,INBF,ETX,CR,LF,MONITOR,ISLID,DUM
0005      INTEGER*2 LTIME,I*RTTM
0006      COMMON /DPM/OUTBF(256),INBF(256),I*RTTM,I*RT,STAT,LTIME
0007      COMMON /GLOB/ ISENT,FL*CNT,IQ*INTH,OUTPCT,IFULL,OUTQ,
1      RESNLM,I*ALTRT,ISLID,MONITOR,LF,ETX,CR,DUM
0008      CC1=INBF(3)
0009      CC2=INBF(4)
0010      IF(CC1 .LT. 0) GOTO 25
0011      IF(CC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0012      IF(CC1 .EQ. 0 .AND. CC2 .EQ. 0) GOTO 40
0013      CC3=MOD(CC1,2)
0014      CC4=MOD(CC1,256)
0015      IF ((CC3 .GE. 1 .OR. CC4 .GE. 128) .AND. ISENT .EQ. 1) GOTO 20
0016      IF ((INBF(1) .EQ. 85) .AND. INBF(2) .EQ. 170) GOTO 99
0017      CC5=MOD(INBF(3),64)
0018      IF(CC5 .GE. 32) CALL INPTQ(LI)
0019      GOTO 99
0020      20      CC7=MOD(CC1,2)
0021      IF(CC7 .GE. 1) GOTO 30
0022      25      CALL ACKNAK(0)
0023      GOTO 99
0024      30      CALL ACKNAK(1)
0025      GOTO 99
0026      40      CALL INPTQ(LI)
0027      99      CONTINUE
0028      RETURN
0029      END
0030
0031
0032
0033
0034

```

```

FORTRAN IV      V02.1-11      PAGE 001

0001      SUBROUTINE INIT
0002      INTEGER*2 XINQ,XOUTQ,ACKQ,PXINQ,PXOUTQ,PACKQ,PINQ,FREE
0003      INTEGER*2 STAT,FLWCNT,OUTFCT
0004      INTEGER*2 SETPRM,RSTPRM,SETBKP,OUTQ,RESNLM,RSTBKP
0005      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDED,OUTRF,INBF
0006      LOGICAL*1 PACK,ETA,CR,LF,MONITOR,ISLID,IUM,IData
0007      LOGICAL*1 SAVE1,SAVE2
0008      INTEGER*2 LTIME,OLITIM,TIMLIM,ACKTIM,ATIMLM,IWRTIM
0009      COMMON /DFM/ OUTRF(256),INBF(256).
0010      1 IWRTIM,IWRT,STAT,LTIME
0011      1 COMMON /QUE/ XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0012      1 COMMON /MESS/ MESSEQ
0013      1 COMMON /BUFS/ XOUTRF(256),XINBF(256),ACKSEQ(256),LIDED(256),
0014      1 IOFLG,LLFLG,IRSEND
0015      COMMON /FRE/ FREE(64),IFR,IFRSZ
0016      COMMON /TIM/ OLDTIM,TIMLIM,ACKTIM,ATIMLM
0017      COMMON /PACK/ PACK(256,64)
0018      COMMON /SWT/ SETPRM,RSTPRM,SETBKP,RSTBKP
0019      COMMON /GLOB/ ISENT,FLWCNT,IQLENTH,OUTFCT,IFULL,OUTQ,
0020      1 RESNLM,IALTRT,ISLID,MONITOR,LF,ETA,CR,DUM
0021      1 COMMON /SCOM/SAVEI(64),SAVE2(64),ITERM,ICOUNT,I1,I2,IData(256),
0022      1 ISWITCH
0023      1 COMMON /SCOM1/ISIM,IVC,IDC,IBC,ICC
0024      CALL RAM(0,2,4)
0025      CALL RAM(0,255,0)
0026      I1=0
0027      I2=0
0028      ICOUNT=0
0029      ISIM=5
0030      IVC=0
0031      IDC=500
0032      IBC=0
0033      ICC=0
0034      MESSEQ=0
0035      IOFLG=0
0036      IRSEND=0
0037      LLFLG=0
0038      IFRSZ=64
0039      INFCT=0
0040      ISLID=28
0041      CP="015
0042      LF="012
0043      ETX="003
0044      RESNLM=2
0045      IQLENTH=16
0046      IALTRT=0
0047      FLWCNT=0
0048      IWRTIM=0
0049      TIMLIM=50
0050      SETPRM=4
0051      RSTPRM=32
0052      SETBKP=8
0053      RSTBKP=C4

```

PAGE 002

```

FORTRAN IV      V02.1-11
0050      ATIMLM=200
0051      MONIOR=25
0052      IWT=0
0053      ISWTC=2
0054      OUTPCT=0
0055      IZULL=0
0056      ISENT=0
0057      DO 40 I=1,20
0058          LIDPD(I)=4
0059          LIDFD(21)=1
0060          LIDFD(22)=3
0061          LIDFD(23)=6
0062          LIDFD(24)=5
0063          LIDFD(25)=7
0064          LIDFD(26)=8
0065          LIDFD(27)=9
0066          LIDFD(28)=2
0067      DO 50 I=29,39
0068          LIDPD(I)=0
0069      DO 60 I=40,44
0070          LIDPD(I)=1
0071      DO 70 I=45,59
0072          LIDPD(I)=0
0073      DO 80 I=60,64
0074          LIDPD(I)=2
0075      DO 90 I=65,79
0076          LIDPD(I)=0
0077      DO 100 I=80,84
0078          LIDPD(I)=4
0079      DO 110 I=85,99
0080          LIDPD(I)=0
0081      DO 120 I=100,104
0082          LIDPD(I)=5
0083      DO 130 I=105,256
0084          LIDPD(I)=0
0085          PINQ(1)=ICLNTH
0086          PINQ(2)=ICLNTH+1
0087          PACKQ(1)=ICLNTH
0088          PACKQ(2)=ICLNTH+1
0089          PIOUTQ(1)=ICLNTH
0090          PIOUTQ(2)=ICLNTH+1
0091          PXINQ(1)=ICLNTH
0092          PXINQ(2)=ICLNTH+1
0093      DO 10 I=1,IPRSZ
0094          PACK(255,I)=0
0095      DO 20 I=1,IPRSZ
0096          PACK(256,I)=0
0097      DO 30 I=1,IPRSZ
0098          FREE(I)=I
0099      DO 30 I=1,256
0100          ACKSEQ(I)=256
0101      IF=IPRSZ
0102      RETURN
0103      END

```


PAGE 001

FORTRAN IV V02.1-11

```

0001 SUBROUTINE LINLOS
0002 REAL*8 RM1(5), LIN08, LIN18
0003 INTEGER*2 FLACNT,OUTFOT,OUTQ,RESNM,T1
0004 INTEGER*2 XINQ,PAIQQ,XOUTQ,ACKQ,PACKQ,PIQQ
0005 LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD
0006 LOGICAL*1 ETX,CR,LF,MONTOR,ISLID,IUM
0007 LOGICAL*1 PACK,M1(40),LIN0(8),LIN1(8)
0008 COMMON/QUE/XINQ(16),PAIQQ(2),XOUTQ(16),PXOUTQ(2),
0009 ACKQ(16),PACKQ(2),INQ(16),PIQQ(2)
0010 & COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0011 & COMMON /GLOB/ ISENT,FLACNT,IQINTH,OUTFOT,IFULL,OUTQ,
0012 1 RESNLM,IALTBT,ISLID,MONTOR,LF,ETX,CR,DUM
0013 COMMON /MESS/ MESSEQ
0014 DATA LIN08,LIN18/'PRIMARY ','BACKUP '//
0015 DATA RM1(1),RM1(2)/'LOSS OF ','MODULATI'//
0016 DATA RM1(3),RM1(4)/'ON ON LO','OP AT NO'//
0017 DATA RM1(5)/'DE 28'//
0018 EQUIVALENCE(LIN0,LIN08)
0019 EQUIVALENCE(LIN1,LIN18)
0020 EQUIVALENCE(M1,RM1)
0021 IS=LLFLG
0022 CALL ENABLE(0)
0023 TI=IGETSP(N)
0024 CALL ENABLE(1)
0025 PACK(1,T1)=0
0026 IF(MESSEQ.EQ.126) MESSEQ=0
0027 MESSEQ=MESSEQ+1
0028 PACK(2,T1)=MESSEQ
0029 PACK(3,T1)=0
0030 PACK(4,T1)=0
0031 PACK(5,T1)=25
0032 PACK(6,T1)=ISLID
0033 DO 20 I=7,9
0034 PACK(I,T1)=LF
0035 CONTINUE
0036 DO 30 I=1,22
0037 PACK(I+9,T1)=M1(I)
0038 IF(IS.EQ.1) GOTO 50
0039 DO 40 I=1,8
0040 PACK(I+31,T1)=LIN0(I)
0041 GOTO 70
0042 DO 60 I=1,8
0043 PACK(I+31,T1)=LIN1(I)
0044 DO 80 I=23,40
0045 PACK(I+17,T1)=M1(I)
0046 PACK(58,T1)=CR
0047 PACK(59,T1)=LF
0048 PACK(60,T1)=ETX
0049 CALL ENSTR(PACK(254,I),60)
0050 CALL ENABLE(0)
0051 CALL ENQUE(PXINQ,XINQ,T1)
0052 CALL ENABLE(1)
0053

```

PAGE 002

FORTRAN IV V02.1-11

0054 LLFIG=0
0055 RETURN
0056 END

PAGE 001

EFCTRAN IV
V02.1-11

EFFECTS IV

```

SUBROUTINE SDCA
LOGICAL*1 XOUTFF,XINBF,ACKSEQ,IIFD,OUTFF,INBF
LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID,LUM
LOGICAL*1 SAVE1,SAVE2,BHEAD(24),IDATA,ENC
LOGICAL*1 XINQ,XOUTQ,ACKQ,FAINQ,FAOUTQ,PACKQ
INTEGER*2 PINQ,K1,K2,K3,AMB(16),AV(16),AD(16)
INTEGER*2 OUTPCT,HESNLM,FLWCNT,OUTQ
INTEGER*4 MESSEQ
INTEGER*4 I14
REAL*8 VMEAS
REAL*4 RHEAD(3),VTRNK,DTRNK,BWID
EQUIVALENCE(RHEAD,BHEAD)
DATA AV/,256,10,10,25,40,40,5,5,10,64,64,64,5,5/
DATA AD/,512,25,25,10,50,50,10,10,128,128,128,10,10/
DATA RHEAD(1),RHEAD(2)/'SWITCH',FIELD//
DATA RHEAD(3)//VALUE//
DATA VTRNK,DTRNK,BWID/'AAAAAAA','BBBBBBB','11110'/
COMMON/MESS/ MESSEC
COMMON/QUE/XINQ(16),PINQ(2),XOUTQ(16),PAOUTQ(2),
& ACKQ(16),PACK(2),INQ(16),PINQ(2)
COMMON/BUFFS/YOUBF(256),XINBF(256),ACKSEQ(255),IIFD(256),
& IOPLG,LJPLG,IRSEND
COMMON/GLOB/ISENT,FLWCNT,IQINTH,OUTPCT,IFULL,OUTQ,
& RESNLM,IALTFT,ISLID,MONTOR,LF,ETX,CR,JUM
COMMON PACK(256,64)
COMMON/SCOM/SAVE1(64),SAVE2(64),ITEM,ICOUNT,I1,I2,IDATA(256),
& ISWTCH
COMMON/SCOM1/ISIM,IVC,IDC,IBC,ICC
COMMON/END /ENC(4)
1 IF(ISWTCH .EQ. 0) GOTO 999
DECODE(4,2,XINBF(1),ERR=999) AVE(1)
DECODE(4,2,XINBF(5),ERR=999) AVE(2)
DECODE(4,2,XINBF(9),ERR=999) AVE(3)
DECODE(4,2,XINBF(13),ERR=999) AVE(4)
DECODE(4,2,XINBF(17),ERR=999) AVE(5)
DECODE(4,2,XINBF(21),ERR=999) AVE(6)
DECODE(4,2,XINBF(25),ERR=999) AVE(7)
DECODE(4,2,XINBF(29),ERR=999) AVE(8)
DECODE(4,2,XINBF(33),ERR=999) AVE(9)
DECODE(4,2,XINBF(37),ERR=999) AVE(10)
DECODE(4,2,XINBF(41),ERR=999) AVE(11)
DECODE(4,2,XINBF(45),ERR=999) AVE(12)
DECODE(4,2,XINBF(49),ERR=999) AVE(13)
DECODE(4,2,XINBF(53),ERR=999) AVE(14)
DECODE(4,2,XINBF(57),ERR=999) AVE(15)
DECODE(4,2,XINBF(61),ERR=999) AVE(16)
FORMAT(I4)
2 IF(AVE(1).EC.2) GOTO 200
C AUTODIN FUNCTION
C
100 DO I=1 J=1,64
200 SAVE1(J)=XINBF(J)
201 CONTINUE

```

PAGE 022

```

FORTRAN IV      V02.1-11
0050      DO 102 J=1,16
0051      JJ=J
0052      IF(AME(J) .LT. AD(J)) GOTO 102
0054      GOTO 300
0055      CONTINUE
0056      GOTO 999
C
C      AUTOVON FUNCTION
C
0057      DO 201 J=1,64
0058      SAVE2(J)=XINBF(J)
0059      CONTINUE
0060      DO 202 J=1,16
0061      JJ=J
0062      IF(AME(J) .LT. AV(J)) GOTO 202
0064      GOTO 300
0065      CONTINUE
0066      GOTO 999
C
C      SEND REPORT TO CRT
C
0067      CALL ENABLE(0)
0068      K1=ICETSP(N)
0069      K2=ICETSP(N)
0070      CALL ENABLE(1)
0071      IF(MESSEQ.EQ. 126) MESSEQ=0
0073      MESSEQ=MESSEQ+1
0074      PACK(1,K1)=0
0075      PACK(2,K1)=MESSEQ
0076      PACK(3,K1)=0
0077      PACK(4,K1)=0
0078      PACK(5,K1)=ITERM
0079      PACK(6,K1)=ISLID
0080      DO 301 J=1,24
0081      PACK(J+6,K1)=HEAD(J)
0082      PACK(31,K1)=CR
0083      PACK(32,K1)=LF
0084      DO 302 J=1,4
0085      PACK(J+32,K1)=XINBF(J)
0086      DO 303 J=1,4
0087      PACK(J+36,K1)="040"
0088      ENCODE(4,304,PACK(41,K1)) JJ
0089      FORMAT(14)
0090      DO 305 J=1,4
0091      PACK(J+44,K1)="240"
0092      JJI=JJ#4
0093      DO 306 J=1,4
0094      PACK(J+48,K1)=XINBF(JJI+J)
0095      PACK(53,K1)=CR
0096      PACK(54,K1)=LF
0097      PACK(55,K1)=EIX
0098      CALL ENSTR(PACK(254,K1),55)
0099      CALL ENABLE(0)
0100      CALL ENQUE(PXINQ,XINQ,K1)

```

PAGE 003

```

FORTRAN IV      V02.1-11
0101      C      CALL ENABLE(1)
          C      SEND REPORT TO DBMS
          C
0102      IF(MESSEQ.EQ.126) MESSEQ=0
0104      MESSEQ=MESSEQ+1
0105      PACK(1,K2)=0
0106      PACK(2,K2)=MESSEQ
0107      PACK(3,K2)=0
0108      PACK(4,K2)=0
0109      PACK(5,K2)=24
0110      PACK(6,K2)=ISLID
0111      PACK(7,K2)=177
0112      DO 307 J=1,4
0113      PACK(J+7,K2)=XINBF(J)
0114      PACK(12,K2)=040
0115      ENCODE(4,304,PACK(13,K2)) JJ
0116      PACK(17,K2)=040
0117      DO 309 J=1,4
0118      PACK(J+17,K2)=XINBF(JJI+J)
0119      PACK(22,K2)=040
0120      PACK(23,K2)=040
0121      PACK(24,K2)=040
0122      PACK(25,K2)=1
0123      PACK(26,K2)=040
0124      PACK(27,K2)=040
0125      PACK(28,K2)=CR
0126      PACK(30,K2)=LF
0127      PACK(29,K2)=ETX
0128      CALL ENSTR(PACK(254,K2),30)
0129      CALL ENABLE(0)
0130      CALL ENQUE(PXINQ,XINQ,K2)
0131      CALL ENABLE(1)
          C
          C      PACKET TO FIAC(RANDOM)
          C
0132      RN=RN(I1,I2)
0133      MON=RN*1000
0134      IF(MON.GE.334.AND.MON.LE.566) GOTO 401
0136      GOTO 999
0137      ISIM=ISIM+2
0138      IF(ISIM.EQ.7) ISIM=1
0140      ICC=ICC+1
0141      IF(ICC.EQ.3) ICC=1
0143      VMEAS=RN(I1,I2)
0144      CALL ENABLE(0)
0145      K3=IGETSP(N)
0146      CALL ENABLE(1)
0147      IF(MESSEQ.EQ.126) MESSEQ=0
0149      MESSEQ=MESSEQ+1
0150      PACK(1,K3)=0
0151      PACK(2,K3)=MESSEQ
0152      PACK(3,K3)=0
0153      PACK(4,K3)=0

```


PAGE 004

```

FORTRAN IV      V02.1-11
0154      PACK(5,K3)=27
0155      PACK(6,K3)=ISLID
0156      IF(ISIM.EQ.3) GOTO 410
0158      IF(ISIM.EQ.5) GOTO 420
0160      ENCODE(8,402,PACK(7,K3)) VTRNK
0161      FORMAT(A8)
0162      IVC=IVC+1
0163      IF(IVC.EQ.501) IVC=1
0165      IS=JICVT(IVC,I14)
0166      CALL IMAGE(I14)
0167      DO 404 J=1,4
0168      404      PACK(J+10,K3)=ENC(J)
0169      ENCODE(2,405,PACK(15,K3)) IVC
0170      405      FORMAT(A2)
0171      GOTO 450
0172      ENCODE(8,402,PACK(7,K3)) DTRNK
0173      IDC=IDC+1
0174      IF(IDC.EQ.1001) IDC=501
0176      IS=JICVT(IDC,I14)
0177      CALL IMAGE(I14)
0178      DO 411 J=1,4
0179      411      PACK(J+10,K3)=ENC(J)
0180      ENCODE(2,405,PACK(15,K3)) IDC
0181      GOTO 450
0182      ENCODE(8,402,PACK(7,K3)) BWID
0183      IBC=IBC+1
0184      IF(IBC.EQ.4) IBC=1
0186      IS=JICVT(IBC,I14)
0187      CALL IMAGE(I14)
0188      PACK(14,K3)=ENC(4)
0189      ENCODE(2,405,PACK(15,K3)) IBC
0190      450      PACK(17,K3)=ICC
0191      ENCODE(4,451,PACK(18,K3)) VMEAS
0192      451      FORMAT(A4)
0193      ENCODE(2,452,PACK(22,K3)) MON
0194      452      FORMAT(A2)
0195      PACK(24,K3)=25
0196      PACK(25,K3)=ISIM
0197      PACK(26,K3)=CR
0198      453      PACK(27,K3)=LF
0199      PACK(28,K3)=ETX
0200      CALL ENSTR(PACK(254,K3),28)
0201      CALL ENABLE(2)
0202      CALL ENQUE(PKING,XING,K3)
0203      CALL ENABLE(1)
0204      999      IOFLG=0
0205      RETURN
0206      END

```

```

FORTRAN IV      V02.1-11      PAGE 001

0001      SUBROUTINE SDCALP
0002      LOGICAL*1 XOUTBF,XINBF,ACKSEQ,LIDFD,OUTBF,INBF
0003      LOGICAL*1 PACK,ETX,CR,LF,MONTOR,ISLID,DUM
0004      LOGICAL*1 SAVE1,SAVE2,IIATA
0005      INTEGER*2 XINQ,XOUTQ,ACKQ,XINQ,PXINQ,PXOUTQ,PACKQ
0006      INTEGER*2 PINQ,K1,OUTFCT,RESNLM,FLWCNT,OUTQ
0007      COMMON/MESSE/MESSEQ
0008      COMMON/CUE/XINQ(16),PXINQ(2),XOUTQ(16),PXOUTQ(2),
0009      & ACKQ(16),PACKQ(2),INQ(16),PINQ(2)
0010      COMMON/BUFS/XOUTBF(256),XINBF(256),ACKSEQ(256),LIDFD(256),
0011      & IOPIC,LLFLG,IRSEND
0012      COMMON/GLOB/ISENT,FLWCNT,IQNTHT,OUTFCT,IFULL,OUTQ,
0013      & RESNLM,IALTHT,ISLID,MONTOR,LF,ETX,CR,DUM
0014      COMMON/SCOM/SAVE1(64),SAVE2(64),ITERM,ICOUNT,I1,I2,IIATA(256),
0015      & ISWTCR
0016      IF(IIATA(1).EQ."117".AND. IIATA(2).EQ."116") GOTO 10
0017      IF(IIATA(1).EQ."117".AND. IIATA(2).EQ."106".AND.
0018      & IIATA(3).EQ."106") GOTO 20
0019      IF(IIATA(1).EQ."115") GOTO 30
0020      IF(IIATA(1).EQ."115") GOTO 30
0021      ISWTCR=1
0022      ITERM=IIATA(4)
0023      RETURN
0024      ISWTCR=0
0025      RETURN
0026      DECODE(4,31,IIATA(2),ERR=999) ITYPE
0027      FORMAT(I)
0028      IF(ITYPE.EQ.1.OR. ITYPE.EQ.2) GOTO 32
0029      GOTO 999
0030      CALL ENABLE(0)
0031      K1=IGETSP(N)
0032      CALL ENABLE(1)
0033      IF(MESSEQ.EQ.126) MESSEQ=0
0034      MESSEQ=MESSEQ+1
0035      PACK(1,K1)=0
0036      PACK(2,K1)=MESSEQ
0037      PACK(3,K1)=0
0038      PACK(4,K1)=0
0039      PACK(5,K1)=25
0040      PACK(6,K1)=28
0041      IF(ITYPE.EQ.2) GOTO 34
0042      I0=35
0043      I0=I0+1,54
0044      PACK(J+0,K1)=SAVE1(J)
0045      CONTINUE
0046      GOTO 40
0047      DO 36 J=1,64
0048      PACK(J+6,K1)=SAVE2(J)
0049      CONTINUE
0050      PACK(71,K1)=CR
0051      PACK(72,K1)=LF
0052      PACK(73,K1)=ETX
0053      CALL ENSTR(PACK(254,K1),73)
0054      CALL ENABLE(0)
0055

```

PAGE 002

```
FORTRAN IV      V02.1-11
0057      CALL ENQUE(PXING,XING,K1)
0058      CALL ENABLE(1)
0059      RETURN
0060      END
```

PAGE 001

```

FORTRAN IV      V02.1-11
0001      SUBROUTINE IMAGE(IN)
0002      INTEGER*4 IN
0003      BYTE ENC
0004      COMMON/END/ENC(4)
0005      ENCODE(4,1,ENC) IN
0006      FORMAT(4I)
0007      DO 2 J=1,4
0008      IF(ENC(J).LE. 57 .AND. ENC(J) .GE. 48) GOTO 2
0009      ENC(J)=0
0010      CONTINUE
0011      RETURN
0012      END
0013

```

```

FDM.MACRO      MACRO V03.02B 01:21:00 PAGE 1

1  .TITLE FDM.MACRO
2  .SETTL NOLE2B
3  .IDENT /V3.0/
4  .GLOBL LIUINT,LIO,ENABLE,SWITCH,MASTER,TIME
5  .GLOBL WOKEN,RAM,STATUS,STATE0
6  .GLOBL LFOUT,ASTART,RUPNT,DESTR,ENSTR,LPIIPT
7  .NLIST CND
8  .PSECT
9
10
11      R0=20
12      R1=21
13      R2=22
14      R3=23
15      R4=24
16      R5=25
17      R6=26
18      R7=27
19
20
21      ;INTERFACE ADDRESS
22      BAR= 172410
23      WCR= 172412
24      CSR= 172414
25      IOBUF= 172416
26      OPREG= 172418
27      HRCR= 177560
28      HRBUF= 177562
29      HXCSR= 177564
30      HXBUF= 177566
31
32      ;CHANGED
33
34      ;INTERFACE OPTIONS
35
36      XCIO=1
37      XSIO=0
38      XGIO=1
39      NDX=0
40      ND24=1
41      NHOST=0
42      DHOST=1
43
44      ;LOCAL VARIABLES
45
46      ARG1= 2
47      ARG2= 4
48      ARG3= 6
49      ARG4= 10
50
51      BIT15= 100000
52      BIT14= 40000
53      BIT08= 400
54      BIT07= 200
55      BIT06= 100
56      BIT05= 40
57      BIT04= 20
58      BIT03= 10
59      BIT02= 4
60
61
62
63
64
65

```


FDM-MACRO
NOTE2B

MACRO V03.02B 01:21:00 PAGE 1-1

66
67
68

000002
000001

BIT01 = 2
BIT00 = 1

MACRO V03.02B 01:21:00 PAGE 2

FDM-MACRO
NODE28

```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****
50 *****
51 *****
52 *****
53 *****
54 *****
55 *****
56 *****
57 *****
58 *****
59 *****
60 *****
61 *****
62 *****
63 *****
64 *****
65 *****
66 *****
67 *****
68 *****
69 *****
70 *****
71 *****
72 *****
73 *****
74 *****
75 *****
76 *****
77 *****
78 *****
79 *****
80 *****
81 *****
82 *****
83 *****
84 *****
85 *****
86 *****
87 *****
88 *****
89 *****
90 *****
91 *****
92 *****
93 *****
94 *****
95 *****
96 *****
97 *****
98 *****
99 *****
100 *****
101 *****

*****
;***** PROCEDURE MASTER START UP *****
;*****
;*****
MASTER: MOV #340, R0 ;PRI=7
MOV R0
MTFS R0
MOV #124, R0 ;LIU HANDLER
MOV #110, (R0)+ ;PRI=7
MOV #340, (R0) ;RESET BUS
RESET
MOV #60, R0 ;SIG INTERFACE HANDLER
MOV #310, (R0)+ ;ENABLE INTERRUPTS
MOV #340, (R0)+ ;CLOCK VECTOR
MOV #100, R0 ;DISABLE INTERRUPTS
MOV #100, R0 ;CLOCK
MOV #TIME, (R0) ;CLEAR LIU
BIC #40100, (R0) ;CLEAR BLUI CSR
JSH PC, LIUINT ;PRI=0
CLR #4CSR, R0 ;CPU DOWN
MOV #000, R0
MTFS R0
RTS PC

*****
;***** PROCEDURE INITIALIZE *****
;*****
;*****
;CLEAR ACRAM
LIUINT: MOV #4354., @OPREG
MOV #2304., @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV #4353., @OPREG
MOV #-256., R0
MOV #2311., @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
INC R0
BNE 1$
;CLEAR INPUT/OUTPUT BUFFERS
MOV #12410, R1
MOV #10440, R0
JSR PC, ZEROBP
MOV #1400, @OPREG
TSTB @CSR
BPL -4
CLRB @CSR
MOV #10510, R1
MOV #10640, R0

;LDADR
;ADDRESS=0
;GOOD WD
;NO RETRY
;SEL ACRAM
;COUNT
;WRITE A NULL
;GOOD WD
;NO LOOP UNTIL READY
;COUNT+1
;RDBUFADR CMD
;SEL INBUF0
;SET POINTER=0
;FALSE READ DATA
;GOOD READ
;NO RETRY
;CLEAR DONE BIT
;RDBUFADR COMMAND
;SEL INBUF1 COMMAND

```

```

102 000234 004767 000052 JSR      ZEROBP
103 000240 012737 001400 MOV      #1400, Q#OPREG
104 000246 105737 172416 TSTB    Q#CSR
105 000252 100375 172414 BPL      -4
106 000254 105037 010510 CLRB     Q#CSR
107 000260 012701 010510 MOV      #10510, R1
108 000264 012700 010540 MOV      #10540, R0
109 000270 004767 000015 JSR      ZEROBP
110 000274 012701 010710 MOV      #10710, R1
111 000300 012700 010740 MOV      #10740, R0
112 000304 004767 000002 JSR      ZEROBP
113 000310 000440 BR        STAT$
114 000312 010137 172416 ZEROBP: MOV  R1, Q#OPREG
115 000316 012737 001400 MOV      #1400, Q#OPREG
116 000324 105737 172414 TSTB    Q#CSR
117 000330 100375 172414 BPL      -4
118 000332 105037 172414 CLRB     Q#CSR
119 000336 013702 172416 MOV      Q#IOBUF, R2
120 000342 042702 177400 BIC      #177400, R2
121 000346 010037 172416 MOV      R0, Q#OPREG
122 000352 022702 000000 CMP      #0, R2
123 000356 001412 BEQ      3$, Q#OPREG
124 000360 012737 001400 MOV      #1400, Q#OPREG
125 000366 105737 172414 TSTB    Q#CSR
126 000372 100375 172414 BPL      -4
127 000374 105037 172414 CLRB     Q#CSR
128 000400 005302 DEC      R2
129 000402 000763 BR        2$, Q#OPREG
130 000404 010037 172416 MOV      R0, PC
131 000410 000207 RTS
132
133
134
135 000412 012737 010400 JSR      READ AND CLEAR STATUS
136 000420 012737 002400 MOV      #4352, Q#OPREG
137 000426 012737 002400 MOV      #1280, Q#OPREG
138 000434 000207 MOV      #1280, Q#OPREG
139
140

```

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

```

*****
;**** PROCEDURE LIU-HANDLER
;*****

```

```

L10:      MOV      R0,      -(SP)
          MOV      R1,      -(SP)
          MOV      R2,      -(SP)
          MOV      R3,      -(SP)
          MOV      R4,      -(SP)
          MOV      R5,      -(SP)
          TST      #0CSR
          BPL      RT1$
          BIC      #BIT14, 0CSR
          MOV      #4352., 0OPREG
          MOV      #1280., 0OPREG
          MOV      0#10PUF, CAUSE
          BIC      #177400, CAUSE

          ;SAVE R0
          ;SAVE R1
          ;SAVE R2
          ;SAVE R3
          ;SAVE R4
          ;SAVE R5
          ;? LIU INTERRUPTED
          ;NO
          ;YES/DISABLE LIU
          ;WCR : RS(0)
          ;RD
          ;CLEAR UNUSED BITS

```

;PERFORM OPERATION DEPENDING ON BITS

```

;OV-FL
;RDBUFADR IN0
;SEL INBUF0
;GO EMPTY
;OV-FL
;RDBUFADR IN1
;SEL INBUF1
;GO EMPTY
;LINE-LOSS PRIMARY
;LATCH ON
;YES LOOP AROUND
;FLAG PRIMARY SW. FAIL
;BACKUP LINE LOSS
;FLAG BACKUP SW. FAIL
;WHITE TOKEN DETECT
;CLEAR TOKEN VAR

```

```

;INBUF0 FULL
;NO
;BUFFER ADDRESS
;RDBUFADR COMM
;SEL BUF
;CRC BIT
;EMPTY BUFFER
;DATA LINK AREA
;1 VARIABLE
;CRC OR BYTE COUNT
;CALL FORTAN QUE'ER

```

EDM-MACRO
NODE28
MACRO V03.02B 01:21:00 PAGE 3-1

```

58 000730 132767 000022 000254 E15: B1B:
59 000736 001423 BEQ
60 000740 012701 000400 MOV #INER, R1
61 000744 012702 010610 MOV #48B., R2
62 000750 012703 010640 MOV #4512., R3
63 000754 012704 000022 MOV #BIT01, R4
64 000760 004757 000046 JSR PC, EMBF
65 000764 012705 001214 MOV #AREA, R5
66 000770 012767 000001 MOV #1, AREA
67 000776 010267 000216 MOV R2, DATA
68 001002 004757 000000 JSR PC, LPINPT
69
70 001006 012605 RTI5:
71 001010 012604 MOV (SP)+, R4
72 001012 012603 MOV (SP)+, R3
73 001014 012602 MOV (SP)+, R2
74 001016 012601 MOV (SP)+, R1
75 001020 012600 MOV (SP)+, R0
76 001022 052737 B1S #BIT14, G#CSR
77 001030 000002 RTI
78
79 001032 012737 012600 172416 EMBF:
80 001040 012737 002400 172416 MOV #48B., G#OPREG
81 001046 013700 172416 MOV #12B0., G#OPREG
82 001052 130400 BITB R4, R0
83 001054 001002 BNE CROCK
84 001056 012704 177777 MOV #C-1, R4
85 001062 010237 172416 MOV R2, G#OPREG
86 001066 012737 001400 MOV #76B., G#OPREG
87 001074 105737 172414 TSTB G#CSR
88 001100 100375 BPL -4
89 001102 013702 172416 MOV #IOBUF, R2
90 001106 042702 177400 BIC #177400, R2
91 001112 010200 MOV R2, R0
92 001114 005400 NEG R0
93 001116 010037 172412 MOV R0, G#WCR
94 001122 010137 172410 MOV R1, G#BAR
95 001126 010337 172416 MOV R3, G#OPREG
96 001132 012737 001400 MOV #76B., G#OPREG
97 001140 105737 172414 TSTB G#CSR
98 001144 100375 BPL -4
99 001146 012737 021000 172416 MOV #8704., G#OPREG
100 001154 000240 NOP
101 001156 105737 TSTB G#CSR
102 001162 100401 BML DMAK
103 001164 000240 NOP
104 001166 012737 004400 172416 DMAK:
105 001174 105737 172414 TSTB G#CSR
106 001200 100375 BPL -4
107 001202 005704 TST R4
108 001204 100001 BPL END5
109 001206 010402 MOV R4, R2
110 001210 000207 RTS
111 001212 000000 END5:
112 001214 000000 CAUSE:
113 001216 001220 AREA:
114 001220 000000 DATA:

```

```

;INBUF1 FULL
;NO MORE DONE
;BUFFER ADDRESS
;RDBUFADP COMM
;SEL INBUF1
;CRC BIT
;GO EMPTY BUFFER
;DATA LINK AREA
;ONE VARIABLE
;CRC OR BYTE COUNT
;CALL FORTRAN QUE'ER
;RESTORE REGISTERS

;ENABLE INTERRUPTS
;RETURN FROM INTERRUPT

;READ STATUS 1
;RS
;FETCH STATUS
;GOOD CRC

;NO FLAG
;RDBUFADR
;RD
;GOOD RD

;REPLACE WITH POINTER
;CLEAR MST BITS
;SAVE IT
;2'S COMP
;BYTE COUNT NOW
;ADDRESS IN MEMORY
;SEL BUFFER
;FALSE RD
;DONE ON
;NO LOOP
;FIRE DMA
;DELAY
;GOOR DMA

;ERROR IF HERE
;FALSE MD
;GOOD MD

;WAS CRC OK
;YES
;NO FLAG IT
;RETURN
;STATUS BYTE 0 HOLDER

```


FDM.MACRO
NODE28
MACRO V03.02B 01:21:00 PAGE 7

```

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
*****
***** PROCEDURE ACRAM
*****
;CALL RAM(0,ADDR,DATA) -WRITE RAM ADDRESS WITH DATA
;CALL RAM(1,ADDR,DATA) -READ RAM DATA AT ADDRESS
;
;0110 =NREAD
;0100 =DREAD
;0111 =NULL
;0000 =WOKEN

;RAM
;WHICH OPERATION
;READ OP
;ADDRESS
;WRITE DATA
;SEL LDATA
;WD/DATA (ADLR)
;WRITE DATA
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;SEL ACRAM
;WD/DATA (CMD)
;WRITE
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;ADDRESS TO READ
;SEL LDATA
;WD/DATA (ADDR)
;WRITE DATA
;VALID WRITE
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;SEL ACRAM
;READ DATA
;VALID READ
;NO LOOP UNTIL READY
;CLEAR DONE BIT
;FETCH DATA
;CLEAR BITS
;RIN DATA

RAM:
MOV GARG1(R5), R0
CMP #0, R0
BNE RDRAM
MOV GARG2(R5), R0
MOV GARG3(R5), R1
MOV #4354., @#OPREG
ADD #2304., R0
MOV R0, @#OPREG
TSTB @#CSR
BPL -4
CLRB @#CSR
MOV #4353., @#OPREG
ADD #2304., R1
MOV R1, @#OPREG
TSTB @#CSR
BPL -4
CLRB @#CSR
RTS PC
RDRAM:
MOV GARG2(R5), R0
MOV #4354., @#OPREG
ADD #2304., R0
MOV R0, @#OPREG
TSTB @#CSR
BPL -4
CLRB @#CSR
MOV #4353., @#OPREG
MOV #768., @#OPREG
TSTB @#CSR
BPL -4
CLRB @#CSR
MOV #1080UF, R0
MOV #177760, R0
BIC R0, GARG3(R5)
MOV R0, R0
RTS PC

```



```

1 *****
2 *****
3 *****
4 *****
5 *****
6 *****
7 *****
8 *****
9 *****
10 *****
11 *****
12 *****
13 *****
14 *****
15 *****
16 *****
17 *****
18 *****
19 *****
20 *****
21 *****
22 *****
23 *****
24 *****
25 *****
26 *****
27 *****
28 *****
29 *****
30 *****
31 *****
32 *****
33 *****
34 *****
35 *****
36 *****
37 *****
38 *****
39 *****
40 *****
41 *****
42 *****
43 *****
44 *****
45 *****
46 *****
47 *****
48 *****
49 *****
50 *****
51 *****
52 *****
53 *****
54 *****
55 *****
56 *****
57 *****
58 *****
59 *****
60 *****
61 *****
62 *****
63 *****
64 *****
65 *****
66 *****
67 *****
68 *****
69 *****
70 *****
71 *****
72 *****
73 *****
74 *****
75 *****
76 *****
77 *****
78 *****
79 *****
80 *****
81 *****
82 *****
83 *****
84 *****
85 *****
86 *****
87 *****
88 *****
89 *****
90 *****
91 *****
92 *****
93 *****
94 *****
95 *****
96 *****
97 *****
98 *****
99 *****
100 *****

```

```

EDM.MACRO
NOI22B
MACRO V03.02B 01:21:00 PAGE 10
*****
;*** PROCEDURE RESTART
;*****
;CALL RSTART(0) -CAUSES A SOFTWARE HALT
;CALL RSTART(1) -RESTARTS PROGRAM (MASTER)
;CALL RSTART(2) -LOAD MODE(173000)
*****
RSTART: MOV GARGI(R5), R0
RS0: CMP #0, R0
BNE RS1
HALT
RTS PC
RS1: CMP #1, R0
BNE RS2
JSR PC, LIUINT
MOV #40, R0
JMP (R0), R0
CMP #2, R0
BNE RST
JMP GROM
RTS PC
ROM: .WORD 173000
;MODE
;LOAD ADDRESS
;RESTART PROGRAM
;INT LIU FIRST
;START ADDRESS
;LOAD MODE
*****
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

```

FDM.MACRO MACRO V03.02B 01:21:00 PAGE 11

NODE28

```

1 002476 01046
2 002500 010146
3 002502 010246
4 002504 042737 000100 177560
5 002512 012700
6 002516 012701 000400
7 002522 105737 177560
8 002526 100375
9 002530 113702 177562
10 002534 005301
11 002536 105737 177560
12 002542 100375
13 002544 113702 177562
14 002550 042702 177400
15 002554 110220
16 002556 005301
17 002560 001366
18 002562 052737 000100 177560
19 002570 012767 000001 002000
20 002576 012602
21 002600 012601
22 002602 012600
23 002604 000002
24 002606 000002
25 002608 000002
26 002610 000002
27 002612 000002
28 002614 000002
29 002616 000002
30 002618 000002
31 002620 000002

;*****.ENABLE LSB*****
;*****PROCEDURE SIG INTERFACE*****
;*****
;*****
SIO:: MOV R0, -(SP) ;SAVE REGISTERS
MOV R1, -(SP)
MOV R2, -(SP)
BIC #100, @#HRCR
MOV #XINBF, R0 ;BUFFER ADDRESS
MOV #67., R1 ;BYTE COUNT
TSTB @#HRCR ;PORT READY
BPL -4 ;LOOP UNTIL
MOV @#HRCR, R2 ;FETCH BYTE
DEC R1 ;COUNT-1
TSTB @#HRCR ;PORT READY
BPL -4 ;LOOP UNTIL
MOV @#HRCR, R2 ;FETCH BYTE
BIC #177400, R2 ;CLEAR MST
MOV R2, (R0)+ ;STORE IT
DEC R1
RNE 25 ;READ 39 BYTES
BIS #100, @#HRCR
MOV #1, IOFLG
MOV (SP)+, R2 ;RETORE REGISTERS
MOV (SP)+, R1
MOV (SP)+, R0
RTI

```

FDM, MACRO
NOTE28

MACRO V03.02B 01:21:00 PAGE 12

```
1
2
3
4
5 000000
6
7 000000
8 000400
9 001000
10 001002
11 001004
12 001006
13
14 000000
15
16
17 000000
18 000400
19 001000
20 001400
21 002000
22 002002
23 002004
24
25
26
27

;*****
;**** COMMON DATA AREA ****
;*****

.PSECT DFM,RW,D,GBL,REL,OVR

OUTBF: .BLKB 256.
INBF: .BLKB 256.
IVRTTM: .BLKW
IVRT: .BLKW
STAT: .BLKW
LTIME: .BLKW
.EVEN

.PSECT BUFS,RW,D,GBL,REL,OVR

XOUTBF: .BLKB 256.
XINBF: .BLKB 256.
ACKSEQ: .BLKB 256.
LIDFD: .BLKB 256.
IOFLG: .BLKW
LFLG: .BLKW
IRSEND: .BLKW
.EVEN

.END

000001
```


F001 001

12:00:00

01-Jul-79

CORRECTION

Program:

decompiler

FOR N0001 /N001 L00
FOR N0001 /N001 L00
FOR S0001 /N001 L00
MAC F001

PAGE 001

12:00:00

01-JUL-79

INSTR.COM

PROGRAM:

DESCRIPTION:

ASSIGN DX0: BK:
R LINK
DX1:ND00 20>DX1:MAP-DX1:ND000 20>T/M
DX1:FLW/C
DX1:ND000 1/E
DX1:SD00 2
\$LMRT

ASSIGN DX1: BK:

1.10 User Language

The User Language provides the human interface to the MSCDM and controls its operation. It is implemented as a dialogue between a processor and a terminal with displays stored as disk files. It is written to handle four ESM-MSCDM terminals. The User Language runs as the application program for node 24 (DBMS) and handles all disk-reads and writes.

1.10.1 Mode 1 CRT-to-CRT

This module allows messages to be sent from the MSCDM terminal to the three ESM terminals. In addition to Mode 1, an ATTACH command is provided so that a terminal can connect to another processor or to another terminal. When a terminal is attached to another terminal it can send messages directly without host processor intervention.

1.10.2 Mode 2 System Inquiry

System inquiry allows the MSCDM user to view displays that define the ESM Multiloop Network Simulation Facility. The default display values which correspond to the parameters generated in the nodal software are stored on the mini-disk. Updates to these parameters are stored on the PDP 11/40 HSTB in Loop 2. Updates are created via Mode 3 (System Update) of the User Language resident on HSTB. Node designators 20-28 are used for loop 5. The LID/FAD Conversion Tables are set up such that the FAD is given for the LID for that location on the display.

1.10.3 Mode 3 - Module Update

This mode of operation is used to modify parameters in the other MSCDM nodes. The user is given a list of parameters that may be changed. The parameter changes are performed in the remote nodes by special control packets that are generated by DBMS.

1.10.4 Mode 4 - File Access

This mode of operation allows records of files to be accessed, modified, added and deleted. The files are the circuit and trunk files.

1.10.5 Mode 5 - Report

This mode of operation is used to generate the system control reports. The operator is led through a menu-selection dialogue. This dialogue first asks whether a file is already open so that the operator can exit from mode 5 and return to complete the report later. Two types of reports can be generated: channel-link and switch. The user is prompted for the various elements of the report and the format of the response. A list of remarks can be added to the report. Simulation of a report being sent to a different System Control hierarchical level via the gateway node is done by entering the node designator of an ESM node. The report can then be sent to an ESM terminal or host processor.

1.10.6 Mode 6 - Status

This mode of operation displays the status of simulated equipment that is being monitored by MSCDM.

1.10.7 Program Descriptions

1.10.7.1 Subroutine USRLNG (FORTRAN):

This subroutine is called by the nodal program when a message is received from the loop. This module checks for abort commands and valid terminal input. The message is then passed to a subroutine of the mode the user requested. It also formats all records read from the disk into packets to be written to the loop.

1.10.7.2 Subroutine WRTDB (FORTRAN):

This subroutine updates records (events) in the status sent by the FIAC node.

1.10.7.3 Subroutine M1000 (FORTRAN):

This subroutine handles all the CRT-to-CRT messages it receives from node 25's LA-36 and reroutes the packet to the designated terminal.

1.10.7.4 Subroutine M2000 (FORTRAN)

This subroutine handles the system inquiries made by the user. It records from the MSG DAT file which contains all the system information and forms it into packets to be sent to the requesting terminal.

1.10.7.5 Subroutine M3000 (FORTRAN)

This subroutine handles the system update for modules VSQC, DSQC, BWBSA, and SDCA. It decodes commands inputted by the user and sends messages to each module concerned.

1.10.7.6 Subroutine M4000 (FORTRAN)

This subroutine handles the file access mode of the User Language. It determines which file (trunk, circuit) is open and which record is currently pointed to.

1.10.7.7 Subroutine M4001 (FORTRAN)

The subroutine is called by subroutine M4000 and performs the editing (addition, deletion, change) of a record in a file (circuit, trunk).

1.10.7.8 Subroutine M5000 (FORTRAN)

This subroutine handles the report mode of the User Language. It fills in a menu of questions and then writes them out to a selected terminal.

1.10.7.9 Subroutine SNDMES (FORTRAN)

This subroutine is called by subroutine M5000 to send the completed report to a terminal in message format.

1.10.7.10 Subroutine M6000 (FORTRAN)

This subroutine handles the status mode of the User Language. It modifies status of a channel or prints the status out to the terminal depending on the response to menu selection.

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:01:42      PAGE 001

0001      SUBROUTINE USRLNG(LI)
0002      LOGICAL*1 F7,F8,F9,F10,F11,F12,IBUFF,TMP(80)
0003      LOGICAL*1 ICODE,IBUF,MODNUM,I7,I8,DUM
0004      LOGICAL*1 PACK,ICON,ETX,LF,CR,MONITOR,ISLID
0005      INTEGER*2 XING,PXING,PING
0006      INTEGER*2 FLWCNT,OUTFCT,OUTQ,RESNM
0007      INTEGER*2 ST,DEVNUM,FTYPE,RECNUM,FREE
0008      COMMON PACK(256,30)
0009      COMMON /MESS/ MESSEQ
0010      COMMON /FRE/ FREE(30),IFR,IFRSZ
0011      COMMON /GLOB/ ISENT,FLWCNT,IOLNTH,OUTFCT,IFULL,OUTQ,
0012      RESNM,IALT,ISLID,MONITOR,LF,ETX,CR,DUM
0013      COMMON /QUE/ XING(15),PXING(2),INQ(15),PING(2)
0014      COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0015      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
0016      1 KEYREC,IBUF(80)
0017      COMMON /M3000/ IFUNT,ITYPE,INODE,IMEAS
0018      COMMON /M4000/ FTYPE,RECNUM,F7,F8,F9,F10,F11,F12
0019      COMMON /M6000/ IDISP,IEQUIP,ICHAN,INUM,IBUFF(20),IFLAG,ICNT
0020      15      FORMAT(1X,80A1)
0021      C *****
0022      C * START DIALOGUE *
0023      C *****
0024      IF(IFIRST.EQ. 1) GOTO 14      !SET AFTER CALLED ONCE
0025      DO 20 IND=1,4
0026      IND=4
0027      DO 21 I=1,256
0028      ICODE(I)=0
0029      GOTO 51
0030      C *****
0031      C * CHECK FOR ABORT *
0032      C *****
0033      14      IF (ICODE(7).EQ. '101'.AND. ICODE(8).EQ. '102'.AND.
0034      ICODE(9).EQ. '117'.AND. ICODE(10).EQ. '122'.AND.
0035      ICODE(11).EQ. '124') GOTO 425      !CHECK FOR ABORT
0036      IF(ICODE(7).NE. '177') GOTO 16
0037      CALL WRTDB
0038      GOTO 12
0039      16      IND=5
0040      MONITOR=ICODE(6)
0041      IF(ICODE(6).EQ. 4) IND=1
0042      IF(ICODE(6).EQ. 8) IND=2
0043      IF(ICODE(6).EQ. 18) IND=3
0044      IF(ICODE(6).EQ. 25) IND=4
0045      IF(IND.EQ. 5) RETURN
0046      C *****
0047      C * CHECK FOR DS *
0048      C *****
0049      IF ((ICODE(7).EQ. '104').AND. (ICODE(8).EQ. '123')) GOTO 18
0050      C *****
0051      C * INTRODUCTION *
0052      C *****
0053      IF(ICODE(7).EQ. 'A') GOTO 600
0054      ITST FOR ATTACH

```

PAGE 002

FORTAN IV V02.1-1 Tue 05-Jun-79 14:01:42

```

0049      GOTO 630
0050      IF (IC0DE(8) .EQ. 'T') GOTO 605
0052      GOTO 630
0053      IF (IC0DE(9) .EQ. 'T') GOTO 610
0055      GOTO 630
0056      IF (IC0DE(10) .EQ. 'A') GOTO 615
0058      GOTO 630
0059      IF (IC0DE(11) .EQ. 'C') GOTO 620
0061      GOTO 630
0062      IF (IC0DE(12) .EQ. 'H') GOTO 625
0064      GOTO 630
0065      CONTINUE
0066      IC0N(7) = ((IC0DE(13)-48) * 10) + (IC0DE(14)-48)
0067      IC0N(8) = *003
0068      IC0N(1) = 85
0069      IC0N(2) = 170
0070      IC0N(3) = 64
0071      IC0N(5) = IC0DE(6)
0072      IC0N(6) = IC0DE(5)
0073      CALL ENSTR(IC0N(254),8)
0074      ICFLG=1
0075      GOTO 18
0076      630      IF (ST(IND,1) .EQ. 0001) GOTO 51
0078      IF (ST(IND,1) .EQ. 0002) GOTO 52
0080      IF (ST(IND,1) .EQ. 0003) GOTO 53
0082      IF (ST(IND,1) .EQ. 0004) GOTO 54
0084      IF (ST(IND,1) .GE. 1000) GOTO 1
0086      GOTO 2
0087      1      IF (ST(IND,1) .LE. 1999) GOTO 61
0089      2      IF (ST(IND,1) .GE. 2000) GOTO 3
0091      GOTO 4
0092      3      IF (ST(IND,1) .LE. 2999) GOTO 62
0094      4      IF (ST(IND,1) .GE. 3000) GOTO 5
0096      GOTO 6
0097      5      IF (ST(IND,1) .LE. 3999) GOTO 63
0099      6      IF (ST(IND,1) .GE. 4000) GOTO 7
0101      GOTO 8
0102      7      IF (ST(IND,1) .LE. 4999) GOTO 64
0104      8      IF (ST(IND,1) .GE. 5000) GOTO 9
0106      GOTO 10
0107      9      IF (ST(IND,1) .LE. 5999) GOTO 65
0109      10     IF (ST(IND,1) .GE. 6000) GOTO 11
0111      GOTO 12
0112      11     IF (ST(IND,1) .LE. 6999) GOTO 66
0114      18     NRCNO=22
0115      NREC=1
0116      ST(IND,1)=0001
0117      GOTO 12
0118      51     NRCNO=1
0119      NREC=2
0120      IFIRST=1
0121      ST(IND,1)=0002
0122      GOTO 12

```

!DISPLAY INTRO MESSAGE
!DISPLAY ENTER USERCODE
!DISPLAY ENTER PASSWORD
!SELECT MODE OF OPERATION

!IF DS WAS ENTERED

!DISPLAY INTRO

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:01:42      PAGE 003

C      *      CHECK USERCODE      *
C      *****
0123 52 IF ((IC0DE(7).EQ. '104 .AND. IC0DE(8).EQ. '101 .AND.
      1 IC0DE(9).EQ. '116) .OR. (IC0DE(7).EQ. '117 .AND.
      2 IC0DE(8).EQ. '127 .AND. IC0DE(9).EQ. '105 .AND.
      3 IC0DE(10).EQ. '116)) GOTO 57 !CHECK FOR USERCODE
0125 IF (IC0DE(7).EQ. '114 .AND. IC0DE(8).EQ. '105 .AND.
      4 IC0DE(9).EQ. '116) GOTO 57
0127 IF (IC0DE(7).EQ. '120 .AND. IC0DE(8).EQ. '105 .AND.
      5 IC0DE(9).EQ. '124 .AND. IC0DE(10).EQ. '105) GOTO 57
0129 IF (IC0DE(7).EQ. '120 .AND. IC0DE(8).EQ. '117 .AND.
      6 IC0DE(9).EQ. '124 .AND. IC0DE(10).EQ. '124 .AND.
      7 IC0DE(11).EQ. '105 .AND. IC0DE(12).EQ. '122) GOTO 57
0131 IF (IC0DE(7).EQ. '122 .AND. IC0DE(8).EQ. '117 .AND.
      8 IC0DE(9).EQ. '131) GOTO 57
      GOTO 18
0133 GOTO 18
0134 57 NRCNO=3
0135 N0REC=1
0136 ST(IND,1)=0003
0137 GOTO 12

C      *****
C      *      CHECK PASSWORD      *
C      *****
0138 53 IF ((IC0DE(7).EQ. '105) .AND.
      1 (IC0DE(8).EQ. '123) .AND.
      2 (IC0DE(9).EQ. '115)) GOTO 58 !CHECK PASSWORD
0140 IF (IC0DE(7).EQ. '114 .AND. IC0DE(8).EQ. '105 .AND.
      3 IC0DE(9).EQ. '107) GOTO 58
0142 IF (IC0DE(7).EQ. '115 .AND. IC0DE(8).EQ. '111 .AND.
      4 IC0DE(9).EQ. '103 .AND. IC0DE(10).EQ. '122 .AND.
      5 IC0DE(11).EQ. '117) GOTO 58
0144 IF (IC0DE(7).EQ. '102 .AND. IC0DE(8).EQ. '117 .AND.
      6 IC0DE(9).EQ. '123 .AND. IC0DE(10).EQ. '123) GOTO 58
0146 IF (IC0DE(7).EQ. '123 .AND. IC0DE(8).EQ. '110 .AND.
      7 IC0DE(9).EQ. '117 .AND. IC0DE(10).EQ. '106 .AND.
      8 IC0DE(11).EQ. '101 .AND. IC0DE(12).EQ. '122) GOTO 58
      GOTO 18
0148 GOTO 18
0149 58 NRCNO=4
0150 N0REC=8
0151 ST(IND,1)=0004
0152 GOTO 12

C      *****
C      *      CHECK MODE SELECTION      *
C      *****
0153 54 MODNUM=IC0DE(7)-48 !WHICH MODE WAS SELECTED
0154 IF (MODNUM.EQ. 1) GOTO 101 !DETERMINE WHICH MODE
0156 IF (MODNUM.EQ. 2) GOTO 102 !OF OPERATION WAS
0158 IF (MODNUM.EQ. 3) GOTO 103 !CHOOSEN
0160 IF (MODNUM.EQ. 4) GOTO 104
0162 IF (MODNUM.EQ. 5) GOTO 105
0164 IF (MODNUM.EQ. 6) GOTO 106
0166 NRCNO=12 !IF INCORRECT NUMBER IS
0167 N0REC=1 !ENTERED - ERROR - RETRY
0168 GOTO 12

```


FORTRAN IV V02.1-1 Tue 05-Jun-79 14:01:42 PAGE 004

!BEGIN CRT-TO-CRT MODE

!BEGIN SYSTEM INQUIRY MODE

!BEGIN MODULE UPDATE MODE

!BEGIN FILE ACCESS MODE

!BEGIN REPORT MODE

!BEGIN STATUS MODE

!ERROR CONDITION

```

0169 MRCNO=13
0170 MOREC=2
0171 ST(IND,1)=1000
0172 GOTO 12
0173 MRCNO=23
0174 MOREC=5
0175 ST(IND,1)=2000
0176 GOTO 12
0177 MRCNO=34
0178 MOREC=3
0179 ST(IND,1)=3000
0180 GOTO 12
0181 MRCNO=49
0182 MOREC=3
0183 ST(IND,1)=4000
0184 GOTO 12
0185 MRCNO=80
0186 MOREC=3
0187 ST(IND,1)=5000
0188 GOTO 12
0189 MRCNO=255
0190 MOREC=5
0191 ST(IND,1)=6000
0192 GOTO 12
0193 CALL M1000(LI)
0194 GOTO 12
0195 CALL M2000
0196 GOTO 12
0197 CALL M3000
0198 GOTO 12
0199 CALL M4000
0200 READ(3,1,ERR=12,END=12)(TMP(I),I=1,80)
0201 READ(5,1,ERR=12,END=12)(TMP(I),I=1,80)
0202 READ(4,1,ERR=12,END=12)(TMP(I),I=1,80)
0203 READ(6,1,ERR=12,END=12)(TMP(I),I=1,80)
0204 GOTO 12
0205 CALL M5000
0206 READ(7,1,ERR=12,END=12)(TMP(I),I=1,80)
0207 GOTO 12
0208 CALL M6000
0209 GOTO 12

C *****
C * WRITE PROCEDURE
C *****
12 IF (MRCNO.EQ. 0) GOTO 450
200 DO 200 J=7,256
    ICODE(J)=0
214 IF (MRCNO.NE. 12) GOTO 210
216 READ(8,12)(ICODE(I),I=9,87)
217 IF (MESSEQ.EQ. 126) MESSEQ=0
219 MESSEQ=MESSEQ+1
220 ICODE(1)=0
221 ICODE(2)=MESSEQ
222 ICODE(3)=0

```

PAGE 005

FORTTRAN IV V02.1-1 Tue 05-Jun-79 14:01:42

```

0223 ICODE(4)=0
0224 ICODE(5)=MONITOR
0225 ICODE(6)=ISLID
0226 ICODE(88)=CR
0227 ICODE(89)=LF
0228 ICODE(90)=ETX
0229 CALL ENSTR(ICODE(254),91)
0230 CALL ENABLE(0)
0231 JSPC=IGETSP(N)
0232 CALL ENABLE(1)
0233 DO 205 I=1,256
0234   PACK(I,JSPC)=ICODE(I)
0235 CONTINUE
0236 CALL ENABLE(0)
0237 CALL ENQUE(PXING,XING,JSPC)
0238 CALL ENABLE(1)
0239 GOTO 450
0240
0241
0242 DO 310 J=1,NOREC
0243   READ(8,'MRCNO,ERR=450')(IBUF(I),I=1,80)
0244   DO 220 I=80,1,-1
0245     IF(IBUF(I).NE.0) GOTO 230
0246   IEND=I
0247 CONTINUE
0248   ICOUNT=IEND+ICUR
0249   IF(ICOUNT.LT.240) GOTO 240
0250   IF(MESSEQ.EQ.126) MESSEQ=0
0251   MESSEQ=MESSEQ+1
0252   ICODE(1)=0
0253   ICODE(2)=MESSEQ
0254   ICODE(3)=0
0255   ICODE(4)=0
0256   ICODE(5)=MONITOR
0257   ICODE(6)=ISLID
0258   ICODE(ICUR+1)=ETX
0259   CALL ENSTR(ICODE(254),ICUR+1)
0260   CALL ENABLE(0)
0261   JSPC=IGETSP(N)
0262   CALL ENABLE(1)
0263   DO 235 I=1,256
0264     PACK(I,JSPC)=ICODE(I)
0265   CONTINUE
0266   CALL ENABLE(0)
0267   CALL ENQUE(PXING,XING,JSPC)
0268   CALL ENABLE(1)
0269   ICUR=6
0270   DO 250 K=1,IEND
0271     ICODE(ICUR+K)=IBUF(K)
0272     ICUR=ICUR+IEND
0273     ICODE(ICUR+1)=CR
0274     ICODE(ICUR+2)=LF
0275     ICUR=ICUR+2
0276   CONTINUE
0277   ICODE(ICUR+1)=ETX
0278
0279

```

PAGE 006

FORTRAN IV U02.1-1 Tue 05-Jun-79 14:01:42

```

0280 CALL ENSTR(ICODE(254),ICUR+1)
0281 IF(MESSEQ.EQ. 126) MESSEQ=0
0283 MESSEQ=MESSEQ+1
0284 ICODE(1)=0
0285 ICODE(2)=MESSEQ
0286 ICODE(3)=0
0287 ICODE(4)=0
0288 ICODE(5)=MONTOR
0289 ICODE(6)=ISLID
0290 IF(ICFLG.EQ. 0) GOTO 320
0292 CALL ENABLE(0)
0293 K1=IGETSP(N)
0294 CALL ENABLE(1)
0295 DO 315 I=1,256
0296   PACK(I,K1)=ICON(I)
0297 CONTINUE
0298 ICFLG=0
0299 CALL ENABLE(0)
0300 CALL ENQUE(PXING,XING,K1)
0301 CALL ENABLE(1)
0302 CALL ENABLE(0)
0303 JSPC=IGETSP(N)
0304 CALL ENABLE(1)
0305 DO 330 J=1,256
0306   PACK(J,JSPC)=ICODE(J)
0307 CONTINUE
0308 DO 340 J=1,256
0309   ICODE(J)=0
0310 CONTINUE
0311 CALL ENABLE(0)
0312 CALL ENQUE(PXING,XING,JSPC)
0313 CALL ENABLE(1)
0314 GOTO 450
0315 IABORT=1
0316 RETURN
0317 500 END

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:02:26      PAGE 001

0001      SUBROUTINE WRITD8
0002      INTEGER*2 ST
0003      LOGICAL*1 ICON,ICON,LBUF(20)
0004      COMMON /USER/ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0005      IF(ICODE(6).EQ. 28) GOTO 300
0007      IF(ICODE(8).NE. 'X') GOTO 10
0009      IF(ICODE(9).NE. 'X') GOTO 10
0011      IF(ICODE(10).NE. 'X') GOTO 10
0013      IF(ICODE(11).EQ. 'X') GOTO 100
0015      10 DECODE(2,30,ICODE(16),ERR=900) ICHAN
0016      30 FORMAT(A2)
0017      DO 40 I=1,20
0018      LBUF(I)=0
0019      IF(ICHAN.LT. 1 .OR. ICHAN.GT. 1014) GOTO 900
0021      READ(2,ICHAN)(LBUF(I),I=1,20)
0022      ENCODE(1,45,LBUF(15)) ICODE(18)
0023      45 FORMAT(I1)
0024      WRITE(2,ICHAN)(LBUF(J),J=1,20)
0025      NRCNO=0
0026      GOTO 999
0027      100 DECODE(2,110,ICODE(11),ERR=900) LNKNUM
0028      DECODE(2,110,ICODE(13),ERR=900) LNKEQP
0029      110 FORMAT(A2)
0030      IF(LNKNUM.EQ. 1) LNUN=0
0032      IF(LNKNUM.EQ. 2) LNUN=3
0034      IF(LNKNUM.EQ. 3) LNUN=6
0036      READ(2,1005+LNUN+LNKEQP,ERR=900,END=900)(LBUF(I),I=1,20)
0037      ENCODE(1,45,LBUF(15)) ICODE(18)
0038      WRITE(2,1005+LNUN+LNKEQP,ERR=900,END=900)(LBUF(I),I=1,20)
0039      NRCNO=0
0040      GOTO 999
0041      300 LIREC=ICODE(11)-48
0042      DO 310 I=1,20
0043      LBUF(I)=0
0044      310 CONTINUE
0045      DO 320 I=1,16
0046      LBUF(I)=ICODE(I+10)
0047      320 CONTINUE
0048      WRITE(2,1003+LIREC,ERR=900,END=900)(LBUF(J),J=1,20)
0049      NRCNO=0
0050      GOTO 999
0051      900 NRCNO=109
0052      NOREC=3
0053      RETURN
0054      999 READ(2,1,ERR=900,END=900)(LBUF(J),J=1,20)
0055      RETURN
0056      STOP
0057      END

```

```

FORTTRAN IV      V02,1-1      Tue 05-Jun-79 14:02:51      PAGE 001

0001 SUBROUTINE M1000(LI)
0002   BYTE ICODE, I7, I8, A7, IBUF, ICON, PACK
0003   LOGICAL*1 ISLID, MONITOR, LF, ETX, CR, DUM
0004   INTEGER ST, DEVNUM, OUTQ, OUTFCT, RESNLM
0005   INTEGER FREE, FLWCNT, XING, PXING
0006   INTEGER PING, KEYREC
0007   COMMON PACK(256,30)
0008   COMMON /MESS/ MESSQ
0009   COMMON /USER/ ST(4,9), IND, ICODE(256), NRCNO, NOREC, ICON(256)
0010   COMMON /FILE/ J1, J2, J3, J4, J5, DEVNUM, IFIRST, ICFLG, IABORT,
0011     & KEYREC, IBUF(80)
0012   COMMON /FREE/ FREE(30), IFR, IFRSZ
0013   COMMON /GLOB/ ISENT, FLWCNT, IOLNTH, OUTFCT, IFULL, OUTQ,
0014     & RESNLM, IALTRT, ISLID, MONITOR, LF, ETX, CR, DUM
0015   COMMON /QUE/ XING(15), PXING(2), INQ(15), PINQ(2)
0016   DATA LEOP, LF, *177, *12/
0017   IF (ST(IND,1) .EQ. 1000) GOTO 14 !VALIDATE ND
0018   IF (ST(IND,1) .EQ. 1003) GOTO 53 !MESSAGE TO SEND
0019   IF (ST(IND,1) .EQ. 1004) GOTO 54 !ASK WHAT TO DO NEXT
0020   *****
0021   C *****
0022   C *****
0023   C *****
0024   C *****
0025   C *****
0026   C *****
0027   C *****
0028   C *****
0029   C *****
0030   C *****
0031   C *****
0032   C *****
0033   C *****
0034   C *****
0035   C *****
0036   C *****
0037   C *****
0038   C *****
0039   C *****
0040   C *****
0041   C *****
0042   C *****
0043   C *****
0044   C *****
0045   C *****
0046   C *****
0047   C *****
0048   C *****
0049   C *****
0050   C *****
0051   C *****
0052   C *****
0053   C *****
0054   C *****
0055   C *****
0056   C *****

14  DECODE (2,10, ICODE(7), ERR=19) A7
10  FORMAT(I)
    IF (A7 .EQ. 4) GOTO 20
    IF (A7 .EQ. 8) GOTO 21
    IF (A7 .EQ. 18) GOTO 22
    IF (A7 .EQ. 25) GOTO 23
17  NRCNO=12
    NOREC=3
    ST(IND,1)=1000
    RETURN
5   NRCNO=13
    NOREC=2
    ST(IND,1)=1000
    RETURN
20  ST(IND,2)=4
    GOTO 7
21  ST(IND,2)=8
    GOTO 7
22  ST(IND,2)=18
    GOTO 7
23  ST(IND,2)=25
    NRCNO=15
    NOREC=1
    ST(IND,1)=1003
    RETURN
53  DO 70 I=1,80
70  IBUF(I)=0
    READ(8,16) (IBUF(I), I=1,24)
    ENCODE(2,100, IBUF(26)) ST(IND,2)
100 FORMAT(I2)
    WRITE(8,16) (IBUF(I), I=1,30)
    NRCNO=16

!LIST CRT ND INFO

!PUT ND INTO ST(IND,2)

!DISPLAY 'TYPE
!IN MESSAGE'

```


PAGE 002

FORTRAN IV V02.1-1 Tue 05-Jun-79 14:02:51

```

0057 NOREC=6
0058 ST(IND,1)=1004
0059 CALL ENABLE(0)
0060 I1=IGTSP(N)
0061 CALL ENABLE(1)
0062 IF(MESSEQ.EQ.126) MESSEQ=0
0064 MESSEQ=MESSEQ+1
0065 PACK(1,1)=0
0066 PACK(2,1)=MESSEQ
0067 PACK(3,1)=0
0068 PACK(4,1)=0
0069 PACK(5,1)=ST(IND,2)
0070 PACK(6,1)=ISLID
0071 IF(ST(IND,2).EQ.25) GOTO 60
0073 PACK(7,1)=*024
0074 DO 61 I=8,23
0075 PACK(I,1)=LF
0076 DO 62 I=1,LI-6
0077 JJ=I
0078 PACK(I+23,1)=ICODE(I+6)
0079 PACK(JJ+25,1)=*024
0080 PACK(JJ+26,1)=*024
0081 PACK(JJ+27,1)=0
0082 PACK(JJ+28,1)=ETX
0083 CALL ENSTR(PACK(254,1),JJ+28)
0084 CALL ENABLE(0)
0085 CALL ENQUE(PXINQ,XINQ,I1)
0086 CALL ENABLE(1)
0087 RETURN
0088 DO 52 I=7,LI
0089 PACK(I,1)=ICODE(I)
0090 CONTINUE
0091 PACK(LI+1,1)=CR
0092 PACK(LI+2,1)=LF
0093 PACK(LI+3,1)=ETX
0094 CALL ENSTR(PACK(254,1),LI+3)
0095 CALL ENABLE(0)
0096 CALL ENQUE(PXINQ,XINQ,I1)
0097 CALL ENABLE(1)
0098 RETURN
C *****
C * WHAT TO DO NEXT *
C *****
0099 54 ANS=ICODE(7)-48
0100 58 IF(ANS.EQ.1) GOTO 700
0102 IF(ANS.EQ.2) GOTO 1
0104 IF(ANS.EQ.3) GOTO 6
0106 IF(ANS.EQ.4) GOTO 8
0108 19 NRCNO=12
0109 NOREC=1
0110 RETURN
0111 700 NRCNO=15
0112 NOREC=1
0113 ST(IND,1)=1003

```

NEW MESSAGE TO SAME CRT

```
FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:02:51      PAGE 003

0114      RETURN
0115      1  NRCNO=13
0116      NOREC=2
0117      ST(IND,1)=1000
0118      RETURN
0119      6  NRCNO=22
0120      NOREC=1
0121      ST(IND,1)=9999
0122      RETURN
0123      8  NRCNO=5
0124      NOREC=7
0125      ST(IND,1)=0004
0126      RETURN
0127      END

                                !NEW MESSAGE TO ANOTHER CRT

                                !LOGOUT

                                !NEW MODE OF OPERATION
```

PAGE 001

Tue 05-Jun-79 14:03:22

V02.1-1

FORTRAN IV

```

0001 SUBROUTINE M2000
0002 REAL*8 MOUT(10,10),LINE(10)
0003 BYTE ND,17,18,ANS,ICODE,ICON
0004 INTEGER ST
0005 COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0006 IF (ST(IND,1) .EQ. 2000) GO TO 39
0007 IF (ST(IND,1) .EQ. 2003) GO TO 53
0008 IF (ST(IND,1) .EQ. 2004) GO TO 54
0009 IF (ST(IND,1) .EQ. 2005) GO TO 55
0010 IF (ST(IND,1) .EQ. 2006) GO TO 25
0011 IF (ST(IND,1) .EQ. 2006) GO TO 25
0012 IF (ST(IND,1) .EQ. 2006) GO TO 25
0013 IF (ST(IND,1) .EQ. 2006) GO TO 25
0014 IF (ST(IND,1) .EQ. 2006) GO TO 25
0015 IF (ST(IND,1) .EQ. 2006) GO TO 25
0016 IF (ST(IND,1) .EQ. 2006) GO TO 25
0017 IF (ST(IND,1) .EQ. 2006) GO TO 25
0018 IF (ST(IND,1) .EQ. 2006) GO TO 25
0019 IF (ST(IND,1) .EQ. 2006) GO TO 25
0020 IF (ST(IND,1) .EQ. 2006) GO TO 25
0021 IF (ST(IND,1) .EQ. 2006) GO TO 25
0022 IF (ST(IND,1) .EQ. 2006) GO TO 25
0023 IF (ST(IND,1) .EQ. 2006) GO TO 25
0024 IF (ST(IND,1) .EQ. 2006) GO TO 25
0025 IF (ST(IND,1) .EQ. 2006) GO TO 25
0026 IF (ST(IND,1) .EQ. 2006) GO TO 25
0027 IF (ST(IND,1) .EQ. 2006) GO TO 25
0028 IF (ST(IND,1) .EQ. 2006) GO TO 25
0029 IF (ST(IND,1) .EQ. 2006) GO TO 25
0030 IF (ST(IND,1) .EQ. 2006) GO TO 25
0031 IF (ST(IND,1) .EQ. 2006) GO TO 25
0032 IF (ST(IND,1) .EQ. 2006) GO TO 25
0033 IF (ST(IND,1) .EQ. 2006) GO TO 25
0034 IF (ST(IND,1) .EQ. 2006) GO TO 25
0035 IF (ST(IND,1) .EQ. 2006) GO TO 25
0036 IF (ST(IND,1) .EQ. 2006) GO TO 25
0037 IF (ST(IND,1) .EQ. 2006) GO TO 25
0038 IF (ST(IND,1) .EQ. 2006) GO TO 25
0039 IF (ST(IND,1) .EQ. 2006) GO TO 25
0040 IF (ST(IND,1) .EQ. 2006) GO TO 25
0041 IF (ST(IND,1) .EQ. 2006) GO TO 25
0042 IF (ST(IND,1) .EQ. 2006) GO TO 25
0043 IF (ST(IND,1) .EQ. 2006) GO TO 25
0044 IF (ST(IND,1) .EQ. 2006) GO TO 25
0045 IF (ST(IND,1) .EQ. 2006) GO TO 25
0046 IF (ST(IND,1) .EQ. 2006) GO TO 25
0047 IF (ST(IND,1) .EQ. 2006) GO TO 25
0048 IF (ST(IND,1) .EQ. 2006) GO TO 25
0049 IF (ST(IND,1) .EQ. 2006) GO TO 25
0050 IF (ST(IND,1) .EQ. 2006) GO TO 25
0051 IF (ST(IND,1) .EQ. 2006) GO TO 25
0052 IF (ST(IND,1) .EQ. 2006) GO TO 25
0053 IF (ST(IND,1) .EQ. 2006) GO TO 25
0054 IF (ST(IND,1) .EQ. 2006) GO TO 25
0055 IF (ST(IND,1) .EQ. 2006) GO TO 25

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:03:22      PAGE 002

0054      IF(ST(IND,3) .EQ. 3) IREC=31
0058      GOTD 22
0059      IREC=29
0060      IF(ST(IND,3) .EQ. 3) IREC=137
0062      GOTD (72,51,52,10)ST(IND,3)
0063      51  NRCNO=141+IREC
0064      NOREC=7
0065      ST(IND,1)=2004
0066      RETURN
0067      52  NRCNO=176+IREC
0068      NOREC=10
0069      ST(IND,1)=2004
0070      RETURN
0071      10  DO 20 I=1,10
0072      DO 15 J=1,10
0073          MOUT(I,J)=0
0074      CONTINUE
0075      20  CONTINUE
0076      56  DO 57 J=1,10
0077          READ(8,216+J)(MOUT(I,J),I=1,10)INTO MOUT
0078      CONTINUE
0079      DO 30 I=1,10
0080          LINE(I)=0
0081      CONTINUE
0082      READ(8,226+ND)(LINE(I),I=1,10)
0083      MOUT(1,2)=LINE(1)
0084      MOUT(5,2)=LINE(2)
0085      MOUT(8,2)=LINE(3)
0086      DO 58 J=3,9
0087          MOUT(8,J)=LINE(J+1)
0088      CONTINUE
0089      DO 60 J=1,10
0090          WRITE(8,274+J)(MOUT(I,J),I=1,10)
0091      CONTINUE
0092      NRCNO=275
0093      NOREC=10
0094      ST(IND,1)=2004
0095      RETURN
0096      54  NRCNO=30
0097      NOREC=4
0098      ST(IND,1)=2005
0099      RETURN
C *****
C * WHAT TO DO NEXT
C *****
0100      55  ANS=ICD(7)-4B
0101      59  IF(ANS .EQ. 1) GOTD 2
0103      IF(ANS .EQ. 2) GOTD 6
0105      IF(ANS .EQ. 3) GOTD 8
0107      GOTD 71
0108      2   NRCNO=23
0109      NOREC=5
0110      ST(IND,1)=2000
0111      RETURN

```

!INFO FOR LIDS 1-100

!INFO FOR LIDS 101-255

!READ FILE FROM DATA BASE
!INTO MOUT

!MOVE INFO INTO LINE

!ASK WHAT TO DO NEXT

!NEW SYSTEM INQUIRY

PAGE 003

FORTTRAN IV U02.1-1 Tue 05-Jun-79 14:03:22

```

0112      6  MRCNO=22
0113      NOREC=1
0114      ST(IND,1)=9999
0115      RETURN
0116      8  MRCNO=5
0117      NOREC=7
0118      ST(IND,1)=0004
0119      RETURN
0120      71 MRCNO=12
0121      NOREC=1
0122      RETURN
0123      END

```

ILOGOUT

'NEW MODE OF OPERATION


```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:03:51      PAGE 001

0001      SUBROUTINE M3000
0002      LOGICAL*1 ICON,ICON,IBUF
0003      INTEGER ST,XING,PXING,PING,DEVNUM
0004      COMMON /MESS/ MESSER
0005      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
0006      * KEYREC,IBUF(80)
0007      COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0008      COMMON /M3000/ IFUNT,ITYPE,INODE,IMEAS
0009      COMMON/QUE/ XING(15),PXING(2),PING(15),PINQ(2)
0010      IF(ST,IND,1) .EQ. 3000) GOTO 20      !WHAT FUNCTION
0011      IF(ST,IND,1) .EQ. 3005) GOTO 30      !WHAT TYPE
0012      IF(ST,IND,1) .EQ. 3010) GOTO 60      !WHAT NODE
0013      IF(ST,IND,1) .EQ. 3015) GOTO 90      !WHAT MEASUREMENT
0014      IF(ST,IND,1) .EQ. 3020) GOTO 110      !WHAT TO DO NEXT
0015      NRCNO=12
0016      NOREC=1
0017      RETURN
0018
0019      20      IFUNT=ICODE(7)-48      !DETERMINE WHAT FUNCTION WAS
0020      IF(IFUNT .LT. 1 .OR. IFUNT .GT. 4) GOTO 10
0021      IF(IFUNT .EQ. 1) IF=22
0022      IF(IFUNT .EQ. 2) IF=23
0023      IF(IFUNT .EQ. 3) IF=26
0024      IF(IFUNT .EQ. 4) IF=28
0025      NRCNO=37
0026      NOREC=4
0027      ST(IND,1)=3005
0028      RETURN
0029
0030      30      ITYPE=ICODE(7)-48      !DETERMINE WHAT TYPE UPDATE
0031      IF(ITYPE .EQ. 1) GOTO 40
0032      IF(ITYPE .EQ. 2) GOTO 70
0033      IF(ITYPE .EQ. 3) GOTO 50
0034      GOTO 10
0035      NRCNO=41
0036      NOREC=2
0037      ST(IND,1)=3010
0038      RETURN
0039
0040      40      NRCNO=43
0041      NOREC=1
0042      ST(IND,1)=3015
0043      RETURN
0044
0045      50      DECODE(2,62,ICODE(7),ERR=10) INODE      !WHAT NODE TO SEND REPORT TO
0046      62      FORMAT(I)
0047      IF(INODE .NE. 8 .AND. INODE .NE. 4 .AND. INODE .NE. 18
0048      * .AND. INODE .NE. 25) GOTO 10
0049      ICON(7)='0'      !GENERATE ON PACKET
0050      ICON(8)='N'
0051      ICON(10)=INODE
0052      CALL ENSTR(ICON(254),10)
0053      GOTO 80
0054      70      ICON(7)='0'      !GENERATE OFF PACKET
0055      ICON(8)='F'
0056      ICON(9)='F'
0057      CALL ENSTR(ICON(254),9)
0058      IF(MESSEQ .EQ. 126) MESSEQ=0
0059
0060
0061
0062
0063
0064
0065
0066

```

PAGE 002

FORTAN IV V02.1-1 Tue 05-Jun-79 14:03:51

```

0068 MESSEQ=MESSEQ+1
0069 ICON(1)=0
0070 ICON(2)=MESSEQ
0071 ICON(3)=0
0072 ICON(4)=0
0073 ICON(5)=IF
0074 ICON(6)=24
0075 ICFLG=1
0076
0077 90 IF(MESSEQ.EQ.126) MESSEQ=0
0078 MESSEQ=MESSEQ+1
0079
0080 ICON(1)=0
0081 ICON(2)=MESSEQ
0082 ICON(3)=0
0083 ICON(4)=0
0084 ICON(5)=IF
0085 ICON(6)=24
0086 ICON(7)='H'
0087 DO 92 I=7,10
0088 ICON(I+1)=ICODE(I)
0089
0090 92 CONTINUE
0091 DECODE(4,93,ICODE(7),ERR=10) MEAS
0092 FORMAT(14)
0093 IF(IFUNT.EQ.1) GOTO 94
0094 IF(IFUNT.EQ.2) GOTO 95
0095 IF(IFUNT.EQ.3) GOTO 96
0096
0097 94 IF(MEAS.GE.1.AND.MEAS.LE.500) GOTO 98
0098 GOTO 10
0099
0100 95 IF(MEAS.GE.501.AND.MEAS.LE.1000) GOTO 98
0101 GOTO 10
0102
0103 96 IF(MEAS.EQ.1.OR.MEAS.EQ.2.OR.MEAS.EQ.3) GOTO 98
0104 GOTO 10
0105
0106 98 CALL ENSTR(ICON(254),11)
0107 ICFLG=1
0108
0109 100 NRCNO=44
0110 NOREC=5
0111 ST(IND,1)=3020
0112 RETURN
0113
0114 110 INEXT=ICODE(7)-48
0115 IF(INEXT.EQ.1) GOTO 120
0116 IF(INEXT.EQ.2) GOTO 130
0117 IF(INEXT.EQ.3) GOTO 140
0118 IF(INEXT.EQ.4) GOTO 150
0119 IF(INEXT.EQ.5) LOGOFF
0120 GOTO 10
0121
0122 120 NRCNO=34
0123 NOREC=3
0124 ST(IND,1)=3000
0125 RETURN
0126
0127 130 NRCNO=37
0128 NOREC=4
0129 ST(IND,1)=3005
0130 RETURN
0131
0132 140 NRCNO=5
0133 NOREC=7

```

FORTRAN IV V02.1-1 Tue 05-Jun-79 14:03:51 PAGE 003

0133 ST(IND,1)=0004
0134 RETURN
0135 150 NRCNO=22
0136 NOREC=1
0137 ST(IND,1)=9999
0138 RETURN
0139 END

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:44:32      PAGE 001

0001      SUBROUTINE M4000
0002      BYTE ICODE,IBUF(JBUF(80),MTYPE,MOUT,ANS,ICON
0003      BYTE I7,I8,F7,F8,F9,F10,F11,F12
0004      INTEGER ST,RECNUM,FTYPE,DEVNUM
0005      DIMENSION MOUT(80)
0006      COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0007      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
0008      1      KEYREC,IBUF(80)
0009      COMMON /M4000/ FTYPE,RECNUM,F7,F8,F9,F10,F11,F12
0010      DATA IPCNT/'45'/
0011      FORMAT (IX,80A1)
0012      IF(ST(IND,1) .EQ. 4000) GOTO 72      !FIRST TIME IN MODE
0013      IF(ST(IND,1) .EQ. 4002) GOTO 22      !FIND OUT IF TO MODIFY
0014      IF(ST(IND,1) .EQ. 4005) GOTO 30      !ENTER KEY TO MODIFY
0015      IF(ST(IND,1) .EQ. 4006) GOTO 80      !CHECK TO SEE WHAR WAS ENTERED
0016      IF(ST(IND,1) .EQ. 4008) GOTO 61      !KEY WAS NOT FOUND
0017      IF(ST(IND,1) .EQ. 4009) GOTO 65      !DISPLAY SAME RECORD
0018      CALL M4001
0019      RETURN
0020      *****
0021      C      *****
0022      C      *      WHAT FILE WILL BE USED      *
0023      C      *****
0024      72      I7=ICODE(7)-48      !DETERMINE WHAT FILE WILL
0025      IF(I7 .LT. 1 .OR. I7 .GT. 2) GOTO 74
0026      FTYPE=I7
0027      GOTO 75
0028      *****
0029      74      NRCNO=12
0030      NOREC=1
0031      RETURN
0032      *****
0033      C      *****
0034      C      *      DISPLAY BYTE SIZE      *
0035      C      *****
0036      75      READ (8'51,FTYPE,ERR=900,END=900)(JBUF(I),I=1,80)
0037      WRITE(8'54,ERR=900,END=900)(JBUF(I),I=1,80)
0038      NRCNO=54
0039      NOREC=4
0040      ST(IND,1)=4002
0041      RETURN
0042      *****
0043      C      *****
0044      C      *      WILL IT BE MODIFIED      *
0045      C      *****
0046      22      I7=ICODE(7)-48
0047      MTYPE=I7
0048      IF(MTYPE .EQ. 1) GOTO 26
0049      IF(MTYPE .EQ. 2) GOTO 27
0050      GOTO 74
0051      *****
0052      26      NRCNO=58
0053      NOREC=1
0054      ST(IND,1)=4010
0055      RETURN
0056      *****
0057      27      NRCNO=59
0058      NOREC=1
0059      ST(IND,1)=4005
0060      RETURN
0061      *****
0062      *      ACCESS KEY
0063      *****

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:44:32      PAGE 002

0054      F7=IC0DE(7)
0055      F8=IC0DE(8)
0056      F9=IC0DE(9)
0057      F10=IC0DE(10)
0058      F11=IC0DE(11)
0059      F12=IC0DE(12)
0060
28      IF(FTYPE.EQ.1) GOTO 50      !IF CIRCUIT FILE
C      *****
C      * COMPARE TRUNK KEY
C      *****
31      RECNUM=0
      DO 40 J=1,10
      READ (4,J,ERR=900,END=900) (JBUF(I),I=1,80)
      DO 35 K=1,80,8
      RECNUM=RECNUM+1
      IF(JBUF(K).NE.IPCNT) GOTO 33      !CHECK TO SEE IF EOF
      IF(JBUF(K+1).NE.IPCNT) GOTO 33
      IF(JBUF(K+2).EQ.IPCNT) GOTO 60
      IF(F7.NE.JBUF(K)) GOTO 35      !START COMPARING FOR
      IF(F8.NE.JBUF(K+1)) GOTO 35      !TRUNK KEY
      IF(F9.NE.JBUF(K+2)) GOTO 35
      IF(F10.NE.JBUF(K+3)) GOTO 35
      IF(F11.NE.JBUF(K+4)) GOTO 35
      IF(F12.EQ.JBUF(K+5)) GOTO 42
35      CONTINUE
40      GOTO 60
0063      READ (6,RECNUM,ERR=900,END=900) (MOUT(I),I=1,80)
0064
0065      DO 58 J=1,10
0066      RECNUM=RECNUM+1
0067      IF(JBUF(K).NE.IPCNT) GOTO 53
0068      IF(JBUF(K+1).NE.IPCNT) GOTO 53
0069      IF(JBUF(K+2).EQ.IPCNT) GOTO 60
0070      IF(F7.NE.JBUF(K)) GOTO 56      !START COMPARING FOR
0071      IF(F8.NE.JBUF(K+1)) GOTO 56      !CIRCUIT FILE
0072      IF(F9.NE.JBUF(K+2)) GOTO 56
0073      IF(F10.NE.JBUF(K+3)) GOTO 59
0074      CONTINUE
0075      GOTO 60
0076
0077      READ (5,RECNUM,ERR=900,END=900) (MOUT(I),I=1,80)
0078
0079      DO 58 J=1,10
0080      RECNUM=RECNUM+1
0081      IF(JBUF(K).NE.IPCNT) GOTO 53
0082      IF(JBUF(K+1).NE.IPCNT) GOTO 53
0083      IF(JBUF(K+2).EQ.IPCNT) GOTO 60
0084      IF(F7.NE.JBUF(K)) GOTO 56
0085      IF(F8.NE.JBUF(K+1)) GOTO 56
0086      IF(F9.NE.JBUF(K+2)) GOTO 59
0087      CONTINUE
0088      GOTO 60
0089
0090      READ (8,72,ERR=900,END=900) (MOUT(I),I=1,80)
0091      MRCNO=72
0092      MREC=7
0093      ST(IND,1)=4006
      RETURN
C      *****
C      * COMPARE CIRCUIT KEY
C      *****
      !CHECK CIRCUIT FILE
50      RECNUM=0
      DO 58 J=1,10
      READ (3,J,ERR=900,END=900) (JBUF(I),I=1,80)
      DO 56 K=1,80,4
      RECNUM=RECNUM+1
      IF(JBUF(K).NE.IPCNT) GOTO 53      !CHECK FOR EOF
      IF(JBUF(K+1).NE.IPCNT) GOTO 53
      IF(JBUF(K+2).EQ.IPCNT) GOTO 60
      IF(F7.NE.JBUF(K)) GOTO 56
      IF(F8.NE.JBUF(K+1)) GOTO 56
      IF(F9.NE.JBUF(K+2)) GOTO 56
      IF(F10.NE.JBUF(K+3)) GOTO 59
      CONTINUE
      GOTO 60
58      CONTINUE
59      READ (5,RECNUM,ERR=900,END=900) (MOUT(I),I=1,80)
      WRITE (8,72,ERR=900,END=900) (MOUT(I),I=1,80)
      !ASK WHAT TO DO NEXT
      MRCNO=72
0118

```


PAGE 003

FORTRAN IV V02.1-1 Tue 05-Jun-79 14:44:32

```

0119 NOREC=7
0120 ST(IND,1)=4006
0121 RETURN
0122 60 NRCNO=79
0123 NOREC=1
0124 ST(IND,1)=4008
0125 RETURN
0126 61 NRCNO=73
0127 NOREC=6
0128 ST(IND,1)=4006
0129 RETURN
0130 65 I7=ICDCE(7)-48
0131 IF(I7.EQ. 1) GOTO 68
0133 IF(I7.EQ. 2) GOTO 28
0135 GOTO 74
0136 68 ST(IND,1)=4012
0137 CALL M4001
0138 RETURN
C *****
C * WHAT TO DO NEXT *
C *****
0139 80 I7=ICDCE(7)-48
0140 IF(I7.LT. 1 .OR. I7.GT. 5) GOTO 74
0141 ANS=I7
0142 95 IF(ANS.EQ. 1) GOTO 100
0143 IF(ANS.EQ. 2) GOTO 105
0145 IF(ANS.EQ. 3) GOTO 110
0147 IF(ANS.EQ. 4) GOTO 115
0149 IF(ANS.EQ. 5) GOTO 120
0151 GOTO 74
0153 100 NRCNO=55
0154 NOREC=3
0155 ST(IND,1)=4002
0156 RETURN
0157 105 NRCNO=49
0158 NOREC=3
0159 ST(IND,1)=4000
0160 RETURN
0161 110 NRCNO=5
0162 NOREC=7
0163 ST(IND,1)=0004
0164 RETURN
0165 115 NRCNO=22
0166 NOREC=1
0167 ST(IND,1)=9999
0168 RETURN
0169 120 NRCNO=55
0170 NOREC=3
0171 ST(IND,1)=4009
0172 RETURN
0173 900 NRCNO=109
0174 NOREC=3
0175 ST(IND,1)=4019
0176 RETURN
0177

```

!DETERMINE WHAT TO DO NEXT

!NEW RECORD OF SAME FILE

!NEW FILE

!ANOTHER MODE OF OPERATION

!LOGOUT

!SAME RECORD

PAGE 004

Tue 05-Jun-79 14:44:32

V02.1-1

FORTAN IV

END

0178

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:44:57      PAGE 001

0001      SUBROUTINE M4001
0002      BYTE ICON,IBUF(80),IRUF,MOUT,ANS
0003      BYTE I7,I8,F7,F8,F9,F10,F11,F12
0004      INTEGER ST,RECNUM,DEVNUM,FTYPE
0005      DIMENSION MOUT(80)
0006      DATA IPCNT/'45'/
0007      COMMON /USER/ ST(4),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0008      COMMON /FILE/J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
0009      1      KEYREC,IBUF(80)
0010      COMMON /M4000/ FTYPE,RECNUM,F7,F8,F9,F10,F11,F12
0011      FORMAT(1X,80A1)
0012      IF(ST,IND,1) .EQ. 4010) GOTO 50      !ASKED FOR KEY TO BE MODIFIED
0013      IF(ST,IND,1) .EQ. 4012) GOTO 52      !CHECK FOR SAME RECORD
0014      IF(ST,IND,1) .EQ. 4014) GOTO 120      !UPDATE OR DELETE RECORD
0015      IF(ST,IND,1) .EQ. 4016) GOTO 145      !UPDATE FILE
0016      IF(ST,IND,1) .EQ. 4018) GOTO 180      !ADD NEW RECORD
0017      IF(ST,IND,1) .EQ. 4019) GOTO 195      !RESTART
0018      IF(ST,IND,1) .EQ. 4020) GOTO 230      !TYPE IN NEW CIRCUIT RECORD
0019      IF(ST,IND,1) .EQ. 4030) GOTO 270      !TYPE IN NEW TRUNK RECORD
0020      *****
0021      C      * FILE WILL BE MODIFIED *****
0022      C      *****
0023      C      *****
0024      C      *****
0025      C      *****
0026      C      *****
0027      C      *****
0028      C      *****
0029      C      *****
0030      C      *****
0031      C      *****
0032      C      *****
0033      C      *****
0034      C      *****
0035      C      *****
0036      C      *****
0037      C      *****
0038      C      *****
0039      C      *****
0040      C      *****
0041      C      *****
0042      C      *****
0043      C      *****
0044      C      *****
0045      C      *****
0046      C      *****
0047      C      *****
0048      C      *****
0049      C      *****
0050      C      *****
0051      C      *****
0052      C      *****
0053      C      *****
0054      C      *****
0055      C      *****
0056      C      *****
0057      C      *****
0058      C      *****
0059      C      *****
0060      C      *****
0061      C      *****
0062      C      *****

      DO 70 KEYREC=1,10      !START COMPARING CIRCUIT FILE
      READ(3,'KEYREC,ERR=999,END=900')(JBUF(I),I=1,80)
      DO 68 K=1,80,4
      RECNUM=RECNUM+1
      IF(JBUF(K) .NE. IPCNT) GOTO 65
      IF(JBUF(K+1) .NE. IPCNT) GOTO 65
      IF(JBUF(K+2) .EQ. IPCNT) GOTO 115
      IF(F7 .NE. JBUF(K)) GOTO 68
      IF(F8 .NE. JBUF(K+1)) GOTO 68
      IF(F9 .NE. JBUF(K+2)) GOTO 68
      IF(F10 .EQ. JBUF(K+3)) GOTO 72
      CONTINUE
      GOTO 115
      WRITE(5,'RECNUM,ERR=900,END=900')(MOUT(I),I=1,80)
      !FOUND THE KEY
      !ASK TO MODIFY OR DELETE
      NRCNO=60
      NOREC=4
      ST(IND,1)=4014
      RETURN
      *****

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:44:57      PAGE 002

C * COMPARE TRUNK KEY *
C *****
0063 80 RECNUM=0
0064 DO 100 KEYREC=1,10
0065 READ(4,'KEYREC,ERR=900,END=900')(JBUF(I),I=1,80)
0066 DO 96 K=1,80,8
0067 RECNUM=RECNUM+1
0068 IF(JBUF(K).NE. IPCNT) GOTO 88
0069 IF(JBUF(K+1).NE. IPCNT) GOTO 88
0070 IF(JBUF(K+2).EQ. IPCNT) GOTO 115
0071 IF(F7.NE. JBUF(K)) GOTO 96
0072 IF(F8.NE. JBUF(K+1)) GOTO 96
0073 IF(F9.NE. JBUF(K+2)) GOTO 96
0074 IF(F10.NE. JBUF(K+3)) GOTO 96
0075 IF(F11.NE. JBUF(K+4)) GOTO 96
0076 IF(F12.EQ. JBUF(K+5)) GOTO 105
0077 CONTINUE
0078 96 CONTINUE
0079 100 CONTINUE
0080 GOTO 115
0081 105 READ(6,'RECNUM,ERR=900,END=900')(MOUT(I),I=1,80)
0082 WRITE(8,'60,ERR=900,END=900')(MOUT(I),I=1,80)
0083 MRCNO=60
0084 NOREC=4
0085 ST(IND,1)=4014
0086 RETURN
0087 115 MRCNO=67
0088 NOREC=4
0089 ST(IND,1)=4018
0090 RETURN
0091 *****
0092 C * UPDATE OR DELETE *
0093 C *****
0094 120 I7=ICD(7)-48
0095 IF(I7.LT. 1 .OR. I7.GT. 2) GOTO 290
0096 ANS=I7
0097 IF(ANS.EQ. 1) GOTO 140
0098 IF(ANS.EQ. 2) GOTO 160
0099 MRCNO=65
0100 NOREC=2
0101 ST(IND,1)=4016
0102 RETURN
0103 *****
0104 C * WRITE NEW RECORD *
0105 C *****
0106 145 IF(FTYPE.EQ. 2) GOTO 147
0107 WRITE(5,'RECNUM,ERR=900,END=900')(ICD(1),I=7,80)
0108 GOTO 150
0109 147 WRITE(6,'RECNUM,ERR=900,END=900')(ICD(1),I=7,80)
0110 WRITE(8,'72,ERR=900,END=900')(ICD(1),I=7,80)
0111 MRCNO=71
0112 NOREC=8
0113 ST(IND,1)=4006
0114 RETURN
0115 160 IF(FTYPE.EQ. 2) GOTO 170

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:44:57      PAGE 003

0123      JBUF(K)=0
0124      JBUF(K+1)=0
0125      JBUF(K+2)=0
0126      JBUF(K+3)=0
0127      WRITE(3,'KEYREC,ERR=900,END=900')(JBUF(I),I=1,80)
0128      DO 162 I=1,80
0129          JBUF(I)=0
0130      CONTINUE
0131      GOTO 174
0132      JBUF(K)=0
0133      JBUF(K+1)=0
0134      JBUF(K+2)=0
0135      JBUF(K+3)=0
0136      JBUF(K+4)=0
0137      JBUF(K+5)=0
0138      WRITE(4,'KEYREC,ERR=900,END=900')(JBUF(I),I=1,80)
0139      DO 172 I=1,80
0140          JBUF(I)=0
0141      CONTINUE
0142      WRITE(8,'72,ERR=900,END=900')(JBUF(I),I=1,80)
0143      NRCNO=71
0144      NOREC=8
0145      ST(IND,1)=4006
0146      RETURN
C *****
C * ADD A NEW RECORD *
C *****
180      I7=ICDCE(7)-48
181      IF(I7.LT.1.OR.I7.GT.2) GOTO 290
182      ANS=17
183      IF(ANS.EQ.1) GOTO 200
184      NRCNO=73
185      NOREC=6
186      ST(IND,1)=4006
187      RETURN
188      IF(ITYPE.EQ.2) GOTO 240
C *****
C * ADD CIRCUIT FILE *
C *****
189      RECNUM=0
190      DO 215 J=1,10
191          READ(3,'J,ERR=900,END=900')(JBUF(I),I=1,80)
192          DO 210 K=1,80,4
193              RECNUM=RECNUM+1
194              IF(JBUF(K).NE.0) GOTO 205
195              IF(JBUF(K+1).NE.0) GOTO 205
196              IF(JBUF(K+2).NE.0) GOTO 205
197              IF(JBUF(K+3).EQ.0) GOTO 220
198              IF(JBUF(K).NE.IPCNT) GOTO 210
199              IF(JBUF(K+1).NE.IPCNT) GOTO 210
200              IF(JBUF(K+2).EQ.IPCNT) GOTO 225
201      CONTINUE
202      CONTINUE
203      JBUF(K)=F7

```



```

0181 JBUF(K+1)=F8
0182 JBUF(K+2)=F9
0183 JBUF(K+3)=F10
0184 224 WRITE(3,J,ERR=900,END=900)(JBUF(I),I=1,80)
0185 NRCNO=64 !TYPE IN NEW CIRCUIT RECORD
0186 NREC=1
0187 ST(IND,1)=4020
0188 RETURN
0189 JBUF(K)=F7
0190 JBUF(K+1)=F8
0191 JBUF(K+2)=F9
0192 JBUF(K+3)=F10
0193 JBUF(K+4)=IPCNT
0194 JBUF(K+5)=IPCNT
0195 JBUF(K+6)=IPCNT
0196 GOTO 224
0197 230 WRITE(5'RECNUM,ERR=900,END=900)(ICODE(I),I=7,80)
0198 GOTO 280
C *****
C * ADD TRUNK FILE *
C *****
0199 240 RECNUM=0 !ADD NEW TRUNK KEY
0200 DO 255 J=1,10 !FINK FIRST EMPTY KEY TO
0201 READ(4,J,ERR=900,END=900)(JBUF(I),I=1,80)
0202 DO 250 K=1,80,8
0203 RECNUM=RECNUM+1
0204 IF(JBUF(K).NE.0) GOTO 245
0205 IF(JBUF(K+1).NE.0) GOTO 245
0206 IF(JBUF(K+2).NE.0) GOTO 245
0207 IF(JBUF(K+3).NE.0) GOTO 245
0208 IF(JBUF(K+4).NE.0) GOTO 245
0209 IF(JBUF(K+5).EQ.0) GOTO 240
0210 IF(JBUF(K+6).NE.0) GOTO 240
0211 IF(JBUF(K).NE.0) GOTO 250
0212 IF(JBUF(K+1).NE.0) GOTO 250
0213 IF(JBUF(K+2).EQ.0) GOTO 245
0214 245 CONTINUE
0215 JBUF(K)=F7
0216 JBUF(K+1)=F8
0217 JBUF(K+2)=F9
0218 JBUF(K+3)=F10
0219 JBUF(K+4)=F11
0220 JBUF(K+5)=F12
0221 WRITE(4,J,ERR=900,END=900)(JBUF(I),I=1,80)
0222 NRCNO=64 !TYPE IN NEW TRUNK RECORD
0223 NREC=1
0224 ST(IND,1)=4030
0225 RETURN
0226 JBUF(K)=F7
0227 JBUF(K+1)=F8
0228 JBUF(K+2)=F9
0229 JBUF(K+3)=F10
0230 JBUF(K+4)=F11
0231 JBUF(K+5)=F12
0232 265 JBUF(K)=F7
0233 JBUF(K+1)=F8
0234 JBUF(K+2)=F9
0235 JBUF(K+3)=F10
0236 JBUF(K+4)=F11
0237 JBUF(K+5)=F12
0238 240
0239
0240

```

```

FORTRAN IV      V03.1-1      Tue 05-Jun-79 14:44:57      PAGE 005

0241      JBUF(K+8)=IPCNT
0242      JBUF(K+9)=IPCNT
0243      JBUF(K+10)=IPCNT
0244      GOTO 264
0245      WRITE(6,'RECNUM,ERR=900,END=900')(ICODE(I),I=7,80)
0246      WRITE(8,'72,ERR=900,END=900')(ICODE(I),I=7,80)
0247      NRCNO=71
0248      NOREC=8
0249      ST(IND,1)=4006
0250      RETURN
0251      NRCNO=12
0252      NOREC=1
0253      RETURN
0254      NRCNO=109
0255      NOREC=3
0256      ST(IND,1)=4019
0257      RETURN
0258      END

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 001

0001      SUBROUTINE M5000
0002      BYTE ICODE,I7,I8,RECRUF,IRUF,STORE,LBUF(80),ICON
0003      INTEGER ST,ANS,PARA,DEVNUM,KEYREC
0004      REAL*8 ERECRF,ROUT(10,9)
0005      DIMENSION STORE(80)
0006      DIMENSION ERECRF(10),RECRUF(80)
0007      COMMON /USER/ ST(4,9),IND,ICODE(256),NRCNO,NOREC,ICON(256)
0008      COMMON /FILE/ J1,J2,J3,J4,J5,DEVNUM,IFIRST,ICFLG,IABORT,
0009      1      KEYREC,IRUF(80)
0010      EQUIVALENCE (ERECRF,RECRUF)
0011      IF(ST(IND,1) .EQ. 5000) GOTO 50
0012      IF(ST(IND,1) .EQ. 5002) GOTO 56
0014      IF(ST(IND,1) .EQ. 5004) GOTO 65
0016      IF(ST(IND,1) .EQ. 5005) GOTO 133
0018      IF(ST(IND,1) .EQ. 5006) GOTO 140
0020      IF(ST(IND,1) .EQ. 5007) GOTO 146
0022      IF(ST(IND,1) .EQ. 5008) GOTO 150
0024      IF(ST(IND,1) .EQ. 5012) GOTO 310
0026      IF(ST(IND,1) .EQ. 5014) GOTO 365
0028      IF(ST(IND,1) .EQ. 5016) GOTO 375
0030      IF(ST(IND,1) .EQ. 5018) GOTO 385
0032      IF(ST(IND,1) .EQ. 5020) GOTO 395
0034      IF(ST(IND,1) .EQ. 5022) GOTO 405
0036      IF(ST(IND,1) .EQ. 5024) GOTO 415
0038      IF(ST(IND,1) .EQ. 5026) GOTO 425
0040      IF(ST(IND,1) .EQ. 5028) GOTO 435
0042      IF(ST(IND,1) .EQ. 5030) GOTO 510
0044      IF(ST(IND,1) .EQ. 5032) GOTO 95
0046      IF(ST(IND,1) .EQ. 5034) GOTO 100
0048      IF(ST(IND,1) .EQ. 5036) GOTO 105
0050      IF(ST(IND,1) .EQ. 5038) GOTO 110
0052      IF(ST(IND,1) .EQ. 5040) GOTO 115
0054      IF(ST(IND,1) .EQ. 5042) GOTO 120
0056      IF(ST(IND,1) .EQ. 5044) GOTO 125
0058      IF(ST(IND,1) .EQ. 5046) GOTO 130
0060      IF(ST(IND,1) .EQ. 5100) GOTO 137
0062      RETURN
0063      NRCNO=12
0064      NOREC=1
0065      RETURN
0066      I7=ICODE(7)-48
0067      IF(I7 .LT. 1 .OR. I7 .GT. 2) GOTO 20
0069      ANS=I7
0070      IF(ANS .EQ. 1) GOTO 300
0072      NRCNO=83
0073      NOREC=3
0074      ST(IND,1)=5002
0075      RETURN
0076      I7=ICODE(7)-48
0077      IF(I7 .LT. 1 .OR. I7 .GT. 2) GOTO 20
0079      IUNIT=I7
0080      PARA=0
0081      JT=2
0082      DO 62 I=1,80

```

! ASK IF A FILE ALREADY OPEN

! NO FILE OPEN, SO CREATE A
! NEW CHANNEL OR SWITCH
! REPORT

! DETERMINE WHAT TYPE REPORT

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 002

0083      LBUF(I)=0
0084      DO 64 I=1,4
0085      WRITE(7,'I,ERR=999,END=999')(LBUF(J),J=1,80)
0086      CONTINUE
0087      NRCNO=86
0088      MOREC=4
0089      ST(IND,1)=5004
0090      RETURN
0091      NRCNO=90
0092      MOREC=1
0093      ST(IND,1)=5032
0094      RETURN
0095      DO 96 J=1,3
0096      RECBUF(J)=ICODE(J+6)
0097      CONTINUE
0098      NRCNO=91
0099      MOREC=1
0100      ST(IND,1)=5034
0101      RETURN
0102      RECBUF(9)=ICODE(7)
0103      RECBUF(10)=ICODE(8)
0104      NRCNO=92
0105      MOREC=1
0106      ST(IND,1)=5036
0107      RETURN
0108      DO 106 J=7,10
0109      RECBUF(J+10)=ICODE(J)
0110      CONTINUE
0111      IF(IUNIT.EQ.2) GOTO 107
0112      NRCNO=93
0113      MOREC=1
0114      GOTO 108
0115      NRCNO=123
0116      MOREC=1
0117      ST(IND,1)=5038
0118      RETURN
0119      DO 111 J=7,11
0120      RECBUF(J+18)=ICODE(J)
0121      CONTINUE
0122      IF(IUNIT.EQ.2) GOTO 118
0123      NRCNO=94
0124      MOREC=1
0125      ST(IND,1)=5040
0126      RETURN
0127      DO 116 J=7,9
0128      RECBUF(J+26)=ICODE(J)
0129      CONTINUE
0130      NRCNO=95
0131      MOREC=1
0132      ST(IND,1)=5042
0133      RETURN
0134      DO 121 J=7,10
0135      RECBUF(J+34)=ICODE(J)
0136      CONTINUE
0137
0138

```

!CREATE A CHANNEL REPORT

!MOVE STATION ID TO ARRAY

!MOVE SEQUENCE NUMBER TO ARRAY

!MOVE DATE-TIME TO ARRAY

!MOVE LINK TO ARRAY

!MOVE CHANNEL NUMBER TO ARRAY

!MOVE OUT TIME TO ARRAY

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 003

0139      NRCNO=96
0140      NOREC=1
0141      ST(IND,1)=5044
0142      RETURN
0143      DO 126 J=7,10
0144          RECBUF(J+2)=ICODE(J)
0145      CONTINUE
0146      NRCNO=97
0147      NOREC=1
0148      ST(IND,1)=5046
0149      RETURN
0150      DO 131 J=7,9
0151          RECBUF(J+50)=ICODE(J)
0152      CONTINUE
0153      WRITE(7,1,ERR=900,END=900)(RECBUF(I),I=1,80)
0154      NRCNO=98
0155      NOREC=3
0156      ST(IND,1)=5005
0157      RETURN
0158      IF(ICODE(7) .NE. '77') GO TO 134
0159      IF(ICODE(8) .NE. '77') GO TO 134
0160      IF(ICODE(9) .EQ. '77') GO TO 136
0161      IF(ICODE(7) .NE. 'X') GO TO 135
0162      IF(ICODE(8) .NE. 'X') GO TO 135
0163      IF(ICODE(9) .NE. 'X') GO TO 135
0164      IF(ICODE(10) .EQ. 'X') GO TO 136
0165      IF(ICODE(7) .NE. '77') GO TO 134
0166      IF(ICODE(8) .NE. '77') GO TO 134
0167      IF(ICODE(9) .EQ. '77') GO TO 136
0168      IF(ICODE(7) .NE. 'X') GO TO 135
0169      IF(ICODE(8) .NE. 'X') GO TO 135
0170      IF(ICODE(9) .NE. 'X') GO TO 135
0171      IF(ICODE(10) .EQ. 'X') GO TO 136
0172      WRITE(7,1,ERR=900,END=900)(ICODE(I),I=7,80)
0173      JT=JT+1
0174      NRCNO=0
0175      RETURN
0176      WRITE(7,1,ERR=900,END=900)(ICODE(I),I=7,80)
0177      NRCNO=113
0178      NOREC=3
0179      JT=2
0180      ST(IND,1)=5100
0181      RETURN
0182      IANS=ICODE(7)-48
0183      IF(IANS .LT. 1 .OR. IANS .GT. 2) GO TO 20
0184      IF(IANS .EQ. 2) GO TO 146
0185      NRCNO=116
0186      NOREC=2
0187      ST(IND,1)=5006
0188      RETURN
0189      DECODE(2,141,ICODE(7),ERR=20)17
0190      FORMAT(1)
0191      IF(17 .EQ. 4 .OR. 17 .EQ. 8 .OR. 17 .EQ. 18 .OR.
0192          17 .EQ. 25) GO TO 143
0193      GO TO 20
0194      ANS=17
0195      CALL SNOWES(ANS,IUNIT)
0196      NRCNO=118
0197      NOREC=5
0198      ST(IND,1)=5008
0199      RETURN
0200
0201

```



```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 004

0202 150      I7=IC0DE(7)-48
0203      IF(I7 .LT. 1 .OR. I7 .GT. 4) GOTO 20
0205      JT=2
0206      ANS=I7
0207 160      IF(ANS .EQ. 1) GOTO 165
0209      IF(ANS .EQ. 2) GOTO 170
0211      IF(ANS .EQ. 3) GOTO 175
0213 165      NRCNO=80
0214      NREC=3
0215      ST(IND,1)=5000
0216      RETURN
0217 170      NRCNO=5
0218      NREC=7
0219      ST(IND,1)=0004
0220      RETURN
0221 175      NRCNO=22
0222      NREC=1
0223      ST(IND,1)=9999
0224      RETURN
0225 300      NRCNO=124
0226      NREC=3
0227      ST(IND,1)=5012
0228      RETURN
0229 310      I7=IC0DE(7)-48
0230      IF(I7 .LT. 1 .OR. I7 .GT. 2) GOTO 20
0232      ANS=I7
0233 320      READ(7,1,ERR=900,END=900)(RECBUF(I),I=1,80)
0234      IF(RECBUF(1) .NE. *77) GOTO 325
0236      IF(RECBUF(2) .NE. *77) GOTO 325
0238      IF(RECBUF(3) .EQ. *77) GOTO 360
0240 325      IF(RECBUF(9) .NE. *77) GOTO 330
0242      IF(RECBUF(10) .EQ. *77) GOTO 370
0244 330      IF(RECBUF(17) .NE. *77) GOTO 335
0246      IF(RECBUF(18) .NE. *77) GOTO 335
0248      IF(RECBUF(19) .EQ. *77) GOTO 380
0250 335      IF(ANS .EQ. 2) GOTO 500
0252      IF(RECBUF(25) .NE. *77) GOTO 340
0254      IF(RECBUF(26) .NE. *77) GOTO 340
0256      IF(RECBUF(27) .EQ. *77) GOTO 390
0258      IF(RECBUF(33) .NE. *77) GOTO 345
0260      IF(RECBUF(34) .NE. *77) GOTO 345
0262      IF(RECBUF(35) .EQ. *77) GOTO 400
0264 345      IF(RECBUF(41) .NE. *77) GOTO 350
0266      IF(RECBUF(42) .NE. *77) GOTO 350
0268      IF(RECBUF(43) .EQ. *77) GOTO 410
0270      IF(RECBUF(49) .NE. *77) GOTO 355
0272      IF(RECBUF(50) .EQ. *77) GOTO 420
0274 355      IF(RECBUF(57) .NE. *77) GOTO 356
0276      IF(RECBUF(58) .NE. *77) GOTO 356
0278      IF(RECBUF(59) .EQ. *77) GOTO 430
0280 356      WRITE(7,1,ERR=900,END=900)(RECBUF(I),I=1,80)
0281 357      READ(7,2,ERR=900,END=900)(RECBUF(I),I=1,80)
0282      IF(RECBUF(1) .NE. *77) GOTO 136
0284      IF(RECBUF(2) .NE. *77) GOTO 136

```

! DETERMINE WHAT TO DO NEXT

! NEW REPORT

! NEW MODE OF OPERATION

! LOGOUT

! ASK WHAT TYPE REPORT

! DETERMINE WHAT TYPE REPORT

! CHECK STATION ID

! CHECK SEQUENCE NUMBER

! CHECK DATE-TIME

! CHECK LINK ID

! CHECK CHANNEL NUMBER

! CHECK OUTAGE TIME

! CHECK IN TIME

! CHECK RFD CODE

! CHECK RFD CODE

! CHECK RFD CODE

! CHECK RFD CODE

! CHECK RFD CODE

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 005

0286      IF (RECBUF(3) .EQ. '77') GOTO 450
0288      GOTO 136
0289      MRCNO=90
0290      NOREC=1
0291      ST(IND,1)=5014
0292      RETURN
0293      DO 366 J=1,3
0294      RECBUF(J)=ICCODE(J+6)
0295      CONTINUE
0296      GOTO 325
0297      MRCNO=91
0298      NOREC=1
0299      ST(IND,1)=5016
0300      RETURN
0301      RECBUF(9)=ICCODE(7)
0302      RECBUF(10)=ICCODE(8)
0303      GOTO 330
0304      MRCNO=92
0305      NOREC=1
0306      ST(IND,1)=5018
0307      RETURN
0308      DO 386 J=7,11
0309      RECBUF(J+10)=ICCODE(J)
0310      CONTINUE
0311      GOTO 335
0312      MRCNO=93
0313      NOREC=1
0314      ST(IND,1)=5020
0315      RETURN
0316      DO 396 J=7,11
0317      RECBUF(J+18)=ICCODE(J)
0318      CONTINUE
0319      GOTO 340
0320      MRCNO=94
0321      NOREC=1
0322      ST(IND,1)=5022
0323      RETURN
0324      DO 406 J=7,9
0325      RECBUF(J+26)=ICCODE(J)
0326      CONTINUE
0327      GOTO 345
0328      MRCNO=95
0329      NOREC=1
0330      ST(IND,1)=5024
0331      RETURN
0332      DO 416 J=7,10
0333      RECBUF(J+34)=ICCODE(J)
0334      CONTINUE
0335      GOTO 350
0336      MRCNO=96
0337      NOREC=1
0338      ST(IND,1)=5026
0339      RETURN
0340      DO 426 J=7,10

! MOVE STATION ID TO ARRAY
! MOVE SEQUENCE NUMBER
! MOVE SEQ NUM TO ARRAY
! ENTER DATE-TIME
! MOVE DATE-TIME TO ARRAY
! ENTER LINK ID
! MOVE LINK ID TO ARRAY
! ENTER CHANNEL NUMBER
! MOVE CHAN NUM TO ARRAY
! ENTER OUTAGE TIME
! MOVE OUTAGE TIME TO ARRAY
! ENTER IN TIME

```

```

FORTRAN IV      V02.1-1      Tue 05-Jun-79 14:45:40      PAGE 006

0341      RECBUF(J+42)=ICODE(J)
0342      CONTINUE
0343      GOTO 355
0344      NRCNO=97
0345      NOREC=1
0346      ST(IND,1)=5028
0347      RETURN
0348      DO 436 J=7,10
0349      RECBUF(J+50)=ICODE(J)
0350      CONTINUE
0351      WRITE(7,1,ERR=900,END=900)(RECBUF(I),I=1,80)
0352      GOTO 357
0353      NRCNO=98
0354      NOREC=3
0355      ST(IND,1)=5005
0356      JT=2
0357      RETURN
0358      IF (RECBUF(25) .NE. '77') GOTO 3451CHECK SWITCH ID
0359      IF (RECBUF(26) .NE. '77') GOTO 345
0360      IF (RECBUF(27) .EQ. '77') GOTO 505
0361      GOTO 345
0362      NRCNO=123
0363      NOREC=1
0364      ST(IND,1)=5030
0365      RETURN
0366      DO 511 J=7,11
0367      RECBUF(J+18)=ICODE(J)
0368      CONTINUE
0369      GOTO 410
0370      NRCNO=109
0371      NOREC=3
0372      RETURN
0373      END
0374
0375
0376

```

PAGE 001

FORTAN IV V02.1-1 Tue 05-Jun-79 14:46:20

```

0001 SUBROUTINE SMDMES(LI,LU)
0002 INTEGER*2 XINQ,PXINQ,PINQ
0003 LOGICAL*1 PACK,LBUF(80),RECBUF(80)
0004 COMMON PACK(256,30)
0005 COMMON /MESS/ MESSEQ
0006 COMMON/QUE/XINQ(15),PXINQ(2),INQ(15),PINQ(2)
0007 DO 5 I=1,80
0008   LBUF(I)=0
0009   RECBUF(I)=0
0010   CONTINUE
0011   LCUR=1
0012   READ(7,1,ERR=900,END=900)(RECBUF(I),I=1,80)
0013   IF(LU .EQ. 2) GOTO 100
0014   DO 70 J1=101,108
0015     READ(8,J1,ERR=900,END=900)(LBUF(I),I=1,80)
0016     DO 20 I=80,1,-1
0017       ID=I
0018       IF(LBUF(I) .NE. 0) GOTO 30
0019       CONTINUE
0020       LBUF(ID+1)=*040
0021       DO 40 J=LCUR,LCUR+7
0022         LBUF(ID+J+2-LCUR)=RECBUF(J)
0023         CONTINUE
0024         LBUF(ID+10)=*015
0025         LBUF(ID+11)=*012
0026         LBUF(ID+12)=*003
0027         CALL ENABLE(0)
0028         K1=IGETSP(N)
0029         CALL ENABLE(1)
0030         DO 45 K=1,256
0031           PACK(K,K1)=0
0032           CONTINUE
0033           DO 50 JP=1,ID+12
0034             PACK(JP+6,K1)=LBUF(JP)
0035             CONTINUE
0036             IF(MESSEQ .EQ. 126) MESSEQ=0
0037             MESSEQ=MESSEQ+1
0038             PACK(1,K1)=0
0039             PACK(2,K1)=MESSEQ
0040             PACK(3,K1)=0
0041             PACK(4,K1)=0
0042             PACK(5,K1)=LT
0043             PACK(6,K1)=24
0044             CALL ENSTR(PACK(254,K1),ID+18)
0045             CALL ENABLE(0)
0046             CALL ENQUE(PXINQ,XINQ,K1)
0047             CALL ENABLE(1)
0048             LCUR=LCUR+8
0049             DO 60 I=1,80
0050               LBUF(I)=0
0051               CONTINUE
0052               GOTO 200
0053               DO 190 J1=285,291

```

PAGE 002

F02.1-1 Tue 05-Jun-79 14:46:20

F02.1-1

```

0058 READ(8,'JJ1,ERR=900,END=900')(LBUF(I),I=1,80)
0059 DO 110 I=80,1,-1
0060   ID=I
0061   IF(LBUF(I) .NE. 0) GOTO 120
0063   CONTINUE
0064   LBUF(ID+1)='040
0065   IF(JJ1 .EQ. 289) LCUR=LCUR+8
0067   DO 130 J=LCUR,LCUR+7
0068     LBUF(ID+J+2-LCUR)=RECRUF(J)
0069   CONTINUE
0070   LBUF(ID+10)='015
0071   LBUF(ID+11)='012
0072   LBUF(ID+12)='003
0073   CALL ENABLE(0)
0074   K1=IGETSP(N)
0075   CALL ENABLE(1)
0076   DO 135 K=1,256
0077     PACK(K,K1)=0
0078   CONTINUE
0079   DO 140 JP=1,ID+12
0080     PACK(JP+6,K1)=LBUF(JP)
0081   CONTINUE
0082   IF(MESSEQ .EQ. 126) MESSEQ=0
0084   MESSEQ=MESSEQ+1
0085   PACK(1,K1)=0
0086   PACK(2,K1)=MESSEQ
0087   PACK(3,K1)=0
0088   PACK(4,K1)=0
0089   PACK(5,K1)=LT
0090   PACK(6,K1)=24
0091   CALL ENSTR(PACK(254,K1),ID+18)
0092   CALL ENABLE(0)
0093   CALL ENQUE(PXING,XING,K1)
0094   CALL ENABLE(1)
0095   LCUR=LCUR+8
0096   DO 150 I=1,80
0097     LBUF(I)=0
0098   CONTINUE
0099   CONTINUE
0100   DO 250 IP=2,4
0101     DO 205 I=1,80
0102       LBUF(I)=0
0103   CONTINUE
0104   READ(7,'IP,ERR=900,END=900')(LBUF(I),I=1,80)
0105   IF(LBUF(1) .EQ. '?' .AND. LBUF(2) .EQ. '?') GOTO 260
0107   IF(LBUF(1) .EQ. 'X' .AND. LBUF(2) .EQ. 'X' .AND.
      * LBUF(3) .EQ. 'X') GOTO 260
0109   DO 210 IL=80,1,-1
0110     IF(LBUF(IL) .NE. 0) GOTO 220
0112     CONTINUE
0113     LBUF(IL)='015
0114     LBUF(IL+1)='012
0115     LBUF(IL+2)='003
0116     IF(IL .EQ. 1) GOTO 250

```


PAGE 003

V02,1-1 Tue 05-Jun-79 14:46:20

FORTAN IV

```

0118 CALL ENABLE(0)
0119 K1=IGETSP(N)
0120 CALL ENABLE(1)
0121 DO 225 K=1,256
0122   PACK(K,K1)=0
0123 CONTINUE
0124 DO 230 IR=1,IL+2
0125   PACK(IR+6,K1)=LRUF(IR)
0126 CONTINUE
0127 IF(MESSEQ.EQ.126) MESSEQ=0
0129 MESSEQ=MESSEQ+1
0130 PACK(1,K1)=0
0131 PACK(2,K1)=MESSEQ
0132 PACK(3,K1)=0
0133 PACK(4,K1)=0
0134 PACK(5,K1)=LT
0135 PACK(6,K1)=24
0136 CALL ENSTR(PACK(254,K1),IL+8)
0137 CALL ENABLE(0)
0138 CALL ENQUE(PXING,XING,K1)
0139 CALL ENABLE(1)
0140 CONTINUE
0141 RETURN
0142 NRCNO=109
0143 NOREC=3
0144 RETURN
0145 END

```


PAGE 002

FORTTRAN IV V02.1-1 Tue 05-Jun-79 14:46:53

```

0069 PACK(15,K1)='040
0070 PACK(16,K1)='040
0071 PACK(17,K1)='040
0072 DO 40 I=1,20
0073   PACK(I+17,K1)=IBUFF(I)
0074   40 CONTINUE
0075   PACK(38,K1)='015
0076   PACK(39,K1)='012
0077   PACK(40,K1)='003
0078   CALL ENSTR(PACK(254,K1),40)
0079   CALL ENABLE(0)
0080   CALL ENQUE(PXING,XING,K1)
0081   CALL ENABLE(1)
0082   IFLAG=1
0083   NRCNO=0
0084   GOTO 999
0085   60 IFLAG=0
0086   ST(IND,1)=6010
0087   NRCNO=0
0088   GOTO 999
0089   130 NRCNO=260
0090   NOREC=4
0091   ST(IND,1)=6020
0092   RETURN
0093   IEQUIP=ICODE(7)-48      !WHAT WAS PICKED
0094   IF(IEQUIP.LT.1.OR. IEQUIP.GT.6) GOTO 10
0096   IF(IDISP.EQ.4) GOTO 150
0098   NRCNO=264
0099   NOREC=1
0100   ST(IND,1)=6030
0101   RETURN
0102   150 NRCNO=265
0103   NOREC=2
0104   ST(IND,1)=6030
0105   RETURN
0106   160 DECODE(4,162,ICODE(7),ERR=10)INUM      !GET NUMBER ENTERED
0107   162 FORMAT(I4)
0108   IF(IEQUIP.EQ.1) GOTO 170
0109   IF(IEQUIP.EQ.3) GOTO 175
0110   IF(INUM.LT.1.OR. INUM.GT.3) GOTO 10
0111   IF(IEQUIP.EQ.2) INUM=INUM+1000
0112   IF(IEQUIP.EQ.4) INUM=INUM+1005
0113   IF(IEQUIP.EQ.5) INUM=INUM+1008
0114   IF(IEQUIP.EQ.6) INUM=INUM+1011
0115   GOTO 180
0116   170 IF(INUM.LT.1.OR. INUM.GT.1000) GOTO 10
0117   GOTO 180
0118   175 IF(INUM.EQ.1.OR. INUM.EQ.2) GOTO 177
0119   GOTO 10
0120   177 INUM=INUM+1003
0121   IF(IDISP.EQ.4) GOTO 200
0122   READ(2,INUM,ERR=900,END=900)(IBUFF(J),J=1,20)
0123   WRITE(8,273,ERR=900,END=900)(IBUFF(J),J=1,20)
0124   NRCNO=273

```

PAGE 003

FORTRAN IV V02.1-1 Tue 05-Jun-79 14:46:53

```

0135 NOREC=2
0136 ST(IND,1)=6010
0137 GOTO 999
0138 INO=ICODE(12)-48
0139 IF(INO.LT.1.OR.INO.GT.3) GOTO 10
0141 READ(2,INUM,ERR=900,END=900)(IBUFF(I),I=1,20)
0142 IBUFF(15)=INO*48
0143 WRITE(2,INUM,ERR=900,END=900)(IBUFF(I),I=1,20)
0144 MRCNO=267
0145 NOREC=6
0146 ST(IND,1)=6050
0147 GOTO 999
0148 MRCNO=268
0149 NOREC=5
0150 ST(IND,1)=6050
0151 RETURN
0152 IANS=ICODE(7)-48
0153 IF(IANS.EQ.1) GOTO 320
0155 IF(IANS.EQ.2) GOTO 330
0157 IF(IANS.EQ.3) GOTO 340
0159 IF(IANS.EQ.4) GOTO 350
0161 GOTO 10
0162 ICNT=0
0163 IF(IDISP.EQ.1) GOTO 30
0165 IF(IDISP.EQ.2) GOTO 30
0167 MRCNO=260
0168 NOREC=4
0169 ST(IND,1)=6020
0170 RETURN
0171 MRCNO=5
0172 NOREC=7
0173 ST(IND,1)=0004
0174 RETURN
0175 MRCNO=255
0176 NOREC=5
0177 ST(IND,1)=6000
0178 RETURN
0179 MRCNO=22
0180 NOREC=1
0181 ST(IND,1)=9999
0182 RETURN
0183 MRCNO=109
0184 NOREC=3
0185 ST(IND,1)=6010
0186 RETURN
0187 READ(2,1,ERR=900,END=900)(IBUFF(I),I=1,20)
0188 RETURN
0189 STOP
0190 END

```

1.10.8 MSG.DAT File

This is a listing of the file MSG.DAT which the User Language uses for the menu formats of all five modes.

PAGE 001

11:43:16

23-JUL-79

MSG1.DAT

PROGRAM:

1 THIS IS THE FIM - FEASIBILITY DEVELOPMENT MODEL

2 ENTER USERID PLEASE

3 ENTER PASSWORD PLEASE

4 YOU ARE NOW LOGGED IN - (TO LOGOUT, ENTER *DS*)

5 PLEASE SELECT ONE MODE OF OPERATION:

6 1. CRT TO CRT

7 2. SYSTEM INQUIRY

8 3. MODULE UPDATE

9 4. FILE ACCESS

10 5. REPORT

11 6. STATUS

12 *INVALID ENTRY* - PLEASE TRY AGAIN (OR ENTER DS TO LOGOUT)

13 ENTER DEST CRT NODE DESIGNATOR (ND) - 04 FOR LP#2, 08 FOR LP#3

14 18 FOR LP#4, 25 FOR FIM.

15 PLEASE TYPE IN MESSAGE AND PRESS RETURN

16 MSG SENT TO CRT ND=

17 PLEASE SELECT ONE MODE OF OPERATION:

18 1. NEW MESSAGE TO SAME CRT

19 2. NEW MESSAGE TO ANOTHER CRT

20 3. LOGOUT

21 4. NEW MODE OF OPERATION

22 YOU ARE LOGGED OUT FROM FIM

23 PLEASE SELECT TYPE OF SYSTEM INFORMATION:

24 1. NETWORK DEVICE INFORMATION

25 2. LID/FAD CONVERSION TABLE (LID'S 1-100)

26 3. LID/FAD CONVERSION TABLE (LID'S 101-254)

27 4. WORKPAGE PARAMETERS OF NODES

28 PLEASE ENTER NODE DESIGNATOR (ND)

29 IF ND IS NOT KNOWN, ENTER ND FOR NETWORK DEVICE INFORMATION

30 PLEASE SELECT ONE OF THE FOLLOWING:

31 1. NEW SYSTEM INQUIRY

32 2. LOGOUT

33 3. ANOTHER MODE OF OPERATION

34 PLEASE SELECT COLUMN FUNCTION TO BE UPDATED.

35 1. VSOC 3. RWRSA

36 2. PSOC 4. SICA

37 PLEASE SELECT TYPE OF UPDATE

38 1. EVENT REPORTING ON

39 2. EVENT REPORTING OFF

40 3. MEASUREMENT REQUESTED

41 PLEASE ENTER NODE DESIGNATOR FOR EVENT REPORTS

42 (E.G., 04, 08, 18, 25)

43 PLEASE ENTER CHANNEL, LINK OR SWITCH NUMBER TO BE MEASURED (FORMAT 14)

44 PLEASE SELECT ONE OF THE FOLLOWING:

45 1. NEW COLUMN FUNCTION

46 2. SAME COLUMN FUNCTION

47 3. NEW MODE OF OPERATION

48 4. LOGOUT

49 PLEASE SELECT FILE TO BE ACCESSSED:

50 1. CIRCUIT FILE

51 2. TRUNK FILE

52 THE KEY IS 4 BYTES

53 THE KEY IS 6 BYTES

54 DO YOU WISH TO MODIFY THIS RECORD

55 1. YES

56 2. NO

57

```

58 PLEASE ENTER KEY OF RECORD TO BE MODIFIED
59 PLEASE ENTER ACCESS KEY
60
61 FOR THIS RECORD, PLEASE SELECT TYPE OF DESIRED CHANGE
62 1. UPDATE
63 2. DELETE
64 TYPE IN NEW RECORD
65 MAKE ANY CHANGES YOU WISH USING CRT
66 WHEN CHANGES ARE COMPLETE, PRESS RETURN
67 THE RECORD DOES NOT EXIST
68 DO YOU WISH TO ADD A NEW RECORD TO THE FILE?
69 1. YES
70 2. NO
71 ** MODIFICATION COMPLETE **
72
73 PLEASE SELECT ONE OF THE FOLLOWING:
74 1. NEW RECORD OF FILE
75 2. NEW FILE
76 3. ANOTHER MODE OF OPERATION
77 4. LOGOUT
78 5. SAME RECORD
79 THE RECORD DOES NOT EXIST. PRESS RETURN KEY
80 DO YOU HAVE A FILE ALREADY OPEN?
81 1. YES
82 2. NO
83 PLEASE SELECT TYPE OF REPORT TO BE GENERATED:
84 1. CHANNEL-LINK
85 2. SWITCH
86 IF YOU DO NOT HAVE AN ANSWER FOR A QUESTION
87 ENTER THREE(3) '???'
88 IF THE QUESTION IS NONAPPLICABLE ENTER 'N/A'
89 PRESS RETURN KEY WHEN READY
90 ENTER REPORTING STATION INDICATOR (3 CHARACTERS)
91 ENTER REPORT SEQUENCE NUMBER (2 DIGITS)
92 ENTER DATE-TIME THE REPORT IS MADE (DDTTT)
93 ENTER LINK IDENTIFIER (5 CHARACTERS)
94 ENTER CHANNEL NUMBER (3 DIGITS)
95 ENTER TIME THE OUTAGE BEGAN (TTTT)
96 ENTER TIME THE OUTAGE TERMINATED (TTTT)
97 ENTER REASON FOR OUTAGE CODE (3 CHARACTERS)
98 ENTER REMARKS -
99 MAXIMUM OF THREE LINES
100 USED 'XXXX' TO INDICATE END OF REMARKS:
101 REPORTING STATION INDICATOR
102 REPORT SEQUENCE NUMBER
103 DATE-TIME REPORT IS MADE
104 LINK IDENTIFIER
105 CHANNEL NUMBER
106 TIME THE OUTAGE BEGAN
107 TIME THE OUTAGE TERMINATED
108 REASON FOR OUTAGE CODE
109
110 ** ERROR IN READ/WRITE TO DATA BASE - RECORD NOT UPDATED **
111
112 IS REPORT COMPLETE
113 1. YES
114

```



```

172 1 3 4 5 6 7 8 9 2
173 0
174 0
175 0
176 0
177 -PRESS RETURN FOR NEXT INSTRUCTION-
178 LOOP -1- LID/FAD CONVERSION TABLE, FOR LIDS 101-255
179 0
180 0
181 0
182 0
183 0
184 0
185 0
186 -PRESS RETURN FOR NEXT INSTRUCTION-
187 LOOP -2- LID/FAD CONVERSION TABLE, FOR LIDS 101-255
188 0
189 0
190 0
191 0
192 0
193 0
194 0
195 0
196 -PRESS RETURN FOR NEXT INSTRUCTION-
197 LOOP -3- LID/FAD CONVERSION TABLE, FOR LIDS 101-255
198 0
199 0
200 0
201 0
202 0
203 0
204 0
205 0
206 -PRESS RETURN FOR NEXT INSTRUCTION-
207 LOOP -4- LID/FAD CONVERSION TABLE, FOR LIDS 101-255
208 0
209 0
210 0
211 0
212 0
213 0
214 0
215 0
216 -PRESS RETURN FOR NEXT INSTRUCTION-
217 NODE WORKSPACE PARAMETERS
218 NODE HAS DESIGNATOR IN LOOP#
219 ALTERNATE GATEWAY FUNCTIONAL ADDRESS .....
220 MAXIMUM INPUT QUEUE SIZE (TO EXTERNAL) .....
221 MAXIMUM OUTPUT QUEUE SIZE (TO RTSTREAF) .....
222 MAXIMUM PACKET XMISSIONS BEFORE MSG TERM .....
223 TIMEOUT FOR WRITE TOKEN REGENERATION .....
224 TIMEOUT FOR PACKET RETRANSMISSION .....
225 NUMBER OF NODES IN LOCAL LOOP .....
226 -PRESS RETURN FOR NEXT INSTRUCTION-
227 HSTA 1 1 2+3 10 1 6 41 41 3
228 GATT-2 2 1 2+3 10 1 NA 7 NA 3

```

PAGE 005

11:48:41

23-JUL-79

MSG1.DAT

Program:

decipher

229	GAT1-3	3	1	2,3	10	1	NA	12	NA	3
230	CRT4	4	2	1,3	8	1	8	12	41	4
231	HSIR	5	2	1,3	10	1	6	4	42	4
232	GAT2-1	6	2	1,3	10	1	NA	7	NA	4
233	GAT2-3	7	2	1,3	10	1	NA	6	NA	4
234	CRT8	8	3	1,2	8	1	8	12	41	4
235	GAT3-4	9	3	1,2	10	1	NA	14	NA	4
236	GAT3-2	10	3	1,2	10	1	NA	4	NA	4
237	GAT3-1	11	3	1,2	10	1	NA	7	NA	4
238	GAT4-3	12	4	3,5	10	1	NA	7	NA	8
239	GAT4-5	13	4	3,5	10	1	NA	7	NA	8
240	AUTOIN	14	4	3,5	10	1	NA	7	NA	8
241	TCCF	15	4	3,5	10	1	NA	7	NA	8
242	HSTA	16	4	3,5	10	1	NA	7	NA	8
243	SILC	17	4	3,5	10	1	NA	7	NA	8
244	CRT18	18	4	3,5	10	1	NA	7	NA	8
245	SECUR	19	4	3,5	10	1	NA	7	NA	8
246	SIG	20	NA	NA	NA	NA	NA	NA	NA	NA
247	SSCI	21	5	NA	64	2	4	15	25	8
248	USOC	22	5	NA	64	2	4	15	25	8
249	DSOC	23	5	NA	64	2	4	15	25	8
250	HSID	24	5	NA	64	2	4	15	25	8
251	CRT25	25	5	NA	64	2	4	15	25	8
252	BWBSA	26	5	NA	64	2	4	15	25	8
253	FIAC	27	5	NA	64	2	4	15	25	8
254	SDCA	28	5	NA	64	2	4	15	25	8

PLEASE ENTER ONE OF THE FOLLOWING:

1. DISPLAY RED STATUS

2. DISPLAY AMBER STATUS

3. DISPLAY SELECTED EQUIPMENT

4. MANUALLY CHANGE STATUS OF SELECTED EQUIPMENT

PLEASE SELECT ONE OF THE FOLLOWING:

1. CHANNEL 1-1000

2. LINK 1-3

3. SWITCH 1 OR 2

4. MULTIPLEXOR 1-3

5. TRANSMITTER 1-3

6. RECEIVER 1-3

PLEASE ENTER DEVICE NUMBER TO BE DISPLAYED (FORMAT 14)

PLEASE ENTER DEVICE NUMBER TO BE CHANGED

FOLLOWED BY NEW VALUE (FORMAT 14,11)

MODIFICATION COMPLETE

PLEASE SELECT ONE OF THE FOLLOWING:

1. NEW CHANNEL, LINK, OR SWITCH

2. NEW MODE OF OPERATION

3. SAME MODE OF OPERATION

4. LOGOUT

PRESS RETURN FOR NEXT INSTRUCTION

REPORTING STATION INDICATOR

PAGE 006

11:50:38

23 JUL 79

NGB .DRI

PROGRAM:

downloader

REPORT SEQUENCE NUMBER
 DATE TIME REPORT IS MADE
 SWITCH IDENTIFIER
 CHANNEL NUMBER
 TIME THE OUTAGE BEGAN
 TIME THE OUTAGE TERMINATED
 REASON FOR OUTAGE CODE

286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300
 301
 302
 303
 304
 305
 306
 307
 308
 309
 310
 311
 312
 313
 314
 315
 316
 317
 318
 319
 320
 321
 322

LOOP -5- LID/FAU CONVERSION TABLE, FOR LIDS 101-255

-PRESS RETURN FOR NEXT INSTRUCTION-

PAGE 001

14:56:24

05-JUN-79

COMP2.COM

Program:

decwriter

FOR M0000/NOLINE
FOR M1000/NOLINE
FOR M2000/NOLINE
FOR M3000/NOLINE

PAGE 001

14:56:32

05-JUN-79

COMF2.COM

Program:

decwriter

FOR M0000/NOLINE
FOR M1000/NOLINE
FOR M2000/NOLINE
FOR M3000/NOLINE

1.11 SIG, 11/40 Simulating Input Programs

The Simulated Input Generator (SIG) is a microprocessor that generates simulated inputs to the VSQC, DSQC, and BWBSA modules.

It acts as both a communications sensor and a scanner. It is used to simulate the measurements performed on 1000 channels and three links, multiplexors, transmitters, and receivers. Random numbers are generated for the measurements for the Red, Amber, and Green regions. Approximately 10% of the measurements are in the Amber region and 5% in the Red region. The SIG continually writes to three 9600 baud interfaces to the VSQC, DSQC, and BWBSA nodes. Each channel and link is simulated by a measurement report that is written to the interface. A VSQC measurement report consists of the six parameters given in Section 1.3 plus the channel number, trunk, name, and monitor point. A BWBSA measurement report consists of the thirteen parameters given in Section 1.7 plus the link number and monitor point. In addition, Multiplexor, Transmitter, and Receiver Alarms are generated.

The SIG also responds to commands from the VSQC, DSQC or BWBSA to perform a measurement on a specific channel or link. Equipments which are in the Red region continue to generate Red measurement parameters for a specified number of successive measurements in order to simulate hard equipment failures and repair time.

1.11.1 Program Descriptions

1.11.1.1 Program SIGGEN (FORTRAN)

This program is the simulated input generating program, the program loop that continually sends measurements to VSQC, DSQC, and BWBSA nodes. The function is described in Section 1.11.

1.11.1.2 Subroutine IMAGE (FORTRAN)

This subroutine takes a decimal number and converts it to ASCII with leading zeros placed.

1.11.1.3 Subroutine DELAY (FORTRAN)

This subroutine imposes a delay between measurements.

1.11.1.4 Subroutine INTVEC (MACRO)

This subroutine initializes the interrupt vectors of the interface boards.

1.11.1.5 Subroutine RDVSQC (MACRO)

This subroutine reads messages from the VSQC node.

1.11.1.6 Subroutine RDDSQC (MACRO)

This subroutine reads messages from the DSQC node.

1.11.1.7 Subroutine RDBWSA (MACRO)

This subroutine reads messages from the BWBSA node.

```

0001 PROGRAM SIGGEN
0002 BITE OUTQ(80),ENC
0003 INTEGER*4 K4,I14
0004 INTEGER*2 K2,DFLAG,VFLAG,BFLAG,DGO,VGO,BGO
0005 REAL*4 ALARM,VMEAS,PRED,VLL,BLL,DLL,VFRANGE,DGRANGE,BRANGE
0006 REAL*8 VTRNK,DTRNK,BWID,VTR,BTR
0007 COMMON/EXP/ENC(4)
0008 COMMON/INT/ IFLAG,VFLAG,BFLAG,DGO,VGO,BGO
0009 COMMON/SIG/VMEAS(13),PRED(1023,3),VLL(6),BLL(13,3),DLL,
1 VMEAS(6),BRANGE(13,3),DGRANGE,VTRNK,DTRNK,BWID,
2 ALARM,VTR,BTR
0010 DATA VRANGE/14.0,14.0,160.0,140.0,120.0,150.0/
0011 DATA VLL /-0.5,-0.5,-80.0,-70.0,-60.0,-70.0/
0012 DATA DGRANGE,DLL/30.0,-15.0/
0013 DATA BRANGE/3.0,30.0,3600.0,20.0,100.0,4.0,70.0,3.0,
& 30.0,3600.0,20.0,100.0,30.0,
& 5.0,50.0,3530.0,19.0,91.0,4.0,70.0,6.0,
& 60.0,3530.0,19.0,91.0,27.0,
& 10.0,100.0,3600.0,20.0,100.0,3.9,59.0,
& 10.0,100.0,3600.0,20.0,100.0,25.0/
0014 DATA BLL
& /-1.5,-15.0,0.0,-10.0,-100.0,0.0,-60.0,-1.5,-15.0,
& 0.0,-10.0,-100.0,-15.0,
& -3.0,-30.0,30.0,-9.5,-94.0,0.0,-60.0,-3.0,-30.0,
& 30.0,-9.5,-94.0,-13.5,
& -5.0,-50.0,0.0,-10.0,-100.0,0.1,-59.0,-5.0,-50.0,
& 0.0,-10.0,-100.0,-14.0/
0015 DATA VTRNK,DTRNK,BWID/AAAAAAA',BBB1111', 111110'/
0016 DATA ALARM/XXX'/
0017 DFLAG=0
0018 VFLAG=0
0019 BFLAG=0
0020 VGO=1
0021 DGO=1
0022 BGO=1
0023 I1=0
0024 I2=0
0025 DO 999 J=1,1000
0026 RN=RN+(I1,I2)
999 C
0027 CALL INTVEC
0028 C
0029 C INITIALIZE VECTORS
0030 C
0031 C THESE FLAGS ARE SET FROM A INTERRUPT REQUEST
0032 C -> IF SET PROGRAM JUMPS TO OUTPUT VALUES AND
0033 C THEN RETURNS TO NORMAL LOOP
0034 C
0035 IF(DFLAG.NE.0) GOTO 651
0036 DFLAG=0
0037 IF(VFLAG.NE.0) GOTO 654
0038 VFLAG=0
0039 IF(BFLAG.NE.0) GOTO 657
0040 BFLAG=0
0041 C
0042 C ***** VEQC *****
0043 C

```

PAGE 022

FORTRAN IV V02.1-11

```

0038      IVC=I
0039      DO 350 J=1,6
0040      JJ=J
0041      RN=RN(I1,I2)
0042      IF(RN .GE. .998 .OR. RN .LE. .003) GOTO 400
0044      IF(RN .GE. .990 .OR. RN .LE. .005) GOTO 450
0046      350 VMEAS(J)=RN*VRANGE(J)+VLL(J)
0047      GOTO 500
0048      400 VMEAS(JJ)=RN*VRANGE(JJ)+VLL(JJ)
0049      PRED(IVC,1)=PRED(IVC,1)+1
0050      PRED(IVC,2)=JJ
0051      PRED(IVC,3)=VMEAS(JJ)
0052      600 IF(JJ .EQ. 6) GOTO 500
0054      DO 550 K=JJ+1,6
0055      550 VMEAS(K)=.5*VRANGE(K)+VLL(K)
0056      GOTO 520
0057      450 VMEAS(JJ)=RN*VRANGE(JJ)+VLL(JJ)
0058      GOTO 600
0059      II=I
0060      IS=JICVT(II,I14)
0061      CALL IMAGE(I14)
0062      RN=RN(I1,I2)
0063      MON=RN*1000

      !GREEN
      !RED

      IMON PT

C
C      VSQC REQUEST A INTERRUPT IF FLAG SET
C
0064      IF(VFLAG .EQ. 0) GOTO 659
0066      DO 660 K=1,6
0067      RN=RN(I1,I2)
0068      VMEAS(K)=RN*VRANGE(K)+VLL(K)
0069      IF(PRED(VFLAG,1) .EQ. 0) GOTO 661
0071      IS=JAFIX(PRED(VFLAG,2),K4)
0072      IS=JICVT(K4,K2)
0073      K=K2
0074      VMEAS(K)=PRED(VFLAG,3)
0075      IS=JICVT(VFLAG,I14)
0076      CALL IMAGE(I14)
0077      IVC=VFLAG
0078      RN=RN(I1,I2)
0079      MON=RN*1000
0080      VGO=0
0081      659 CONTINUE

C
0082      IF(VGO .EQ. 1) GOTO 914
0084      ICNT=38
0085      ENCODE(2,601,OUTQ(1)) ICNT
0086      ENCODE(4,501,OUTQ(3)) VMEAS(1)
0087      ENCODE(4,501,OUTQ(7)) VMEAS(2)
0088      ENCODE(4,501,OUTQ(11)) VMEAS(3)
0089      ENCODE(4,501,OUTQ(15)) VMEAS(4)
0090      ENCODE(4,501,OUTQ(19)) VMEAS(5)
0091      ENCODE(4,501,OUTQ(23)) VMEAS(6)
0092      ENCODE(2,502,OUTQ(27)) IVC

```

PAGE 003

```

FORTRAN IV      V02.1-11

0093      ENCODE(9,503,OUTQ(29)) VTRNE
0094      DO 711 J=1,4
0095      OUTQ(J+32)=ENC(J)
0096      ENCODE(2,504,OUTQ(37)) MON
0097      FORMAT(A4)
0098      FORMAT(A2)
0099      FORMAT(A8)
0100      FORMAT(A2)
0101      FORMAT(A2)
D      WRITE(7,900)(VMEAS(10),10=1,5),IVC,
D      (OUTQ(11),10=29,35),MON,
D      900 FORMAT(1X,'WRVSCC->',6F12.6,/,10X,
D      16,2X,8A1,2X,16,/)
D      DO 506 J1=1,40
0102      CALL IFOKEB(177416,OUTQ(J1))
0103      IF(IPEEK(177414).NE. 200) GOTO 505
0104      505 CONTINUE
0106      CALL DELAY
0107      998 CONTINUE
0108      VGO=1
0109      IF(VFLAG.NE. 0) GOTO 655
0110      C *****
C *****
0112      914 IDC=I+500
0113      DO 1350 J=1,3
0114      JJ=J
0115      RN=RN+(11,12)
0116      IF(RN.GE..998 .OR. RN.LE..003) GOTO 1400
0118      IF(RN.GE..990 .OR. RN.LE..005) GOTO 1450
0120      1350 VMEAS(J)=RN*DRANGE+DLL
0121      GOTO 1500
0122      1400 VMEAS(JJ)=RN*DRANGE+DLL
0123      PRED(IDC,1)=PRED(IDC,1)+1
0124      PRED(IDC,2)=JJ
0125      PRED(IDC,3)=VMEAS(JJ)
0126      1600 IF(JJ.EQ. 3) GOTO 1500
0128      DO 1550 K=JJ+1,3
0129      VMEAS(K)=.5*DRANGE+DLL
0130      GOTO 1500
0131      1450 VMEAS(JJ)=RN*DRANGE+DLL
0132      GOTO 1500
0133      1500 11=I+500
0134      IS=JICVT(11,114)
0135      CALL IMAGE(114)
0136      RN=RN+(11,12)
0137      MON=RN+1002
C *****
C *****
C      DSOE REQUESTS A INTERRUPT IF FLAG IS SET
C *****
0138      IF(DFLAG.EQ. 0) GOTO 662
0140      DO 669 K=1,3
0141      RN=RN+(11,12)
0142      VMEAS(K)=RN*DRANGE+DLL

```


PAGE 004

```

PORTMAN IV      V02.1-11
0143 IF(PRED(DFLAG,1).EQ.0) GOTO 662
0145 IS=JAFIX(PRED(DFLAG,2),K4)
0147 IS=IJCVT(K4,K2)
0148 K=K2
0148 VMEAS(K)=PRED(DFLAG,3)
0149 I14=VFLAG-500
0150 IS=JJCVT(I14,I14)
0151 CALL IMAGE(I14)
0152 RN=RN(I1,I2)
0153 MON=RN*1000
0154 DGO=0
0155 662 CONTINUE
C
0156 IF(DGO.EQ.1) GOTO 915
0158 ICNT=26
0159 ENCODE(2,501,OUTQ(1)) ICNT
0160 ENCODE(4,501,OUTQ(3)) VMEAS(1)
0161 ENCODE(4,501,OUTQ(7)) VMEAS(2)
0162 ENCODE(4,501,OUTQ(11)) VMEAS(3)
0163 ENCODE(2,508,OUTQ(15)) IDC
0164 ENCODE(8,509,OUTQ(17)) DTRNK
0165 DO 712 J=1,4
0166 OUTQ(J+20)=ENC(J)
0167 ENCODE(2,510,OUTQ(25)) MON
0168 508 FORMAT(A2)
0169 509 FORMAT(A8)
0170 510 FORMAT(A2)
D
D 511 WRITE(7,901)(VMEAS(IC),IC=1,3),IDC,(OUTQ(ID),ID=17,24),
D 512 MON
D 901 FORMAT(IX,'WRDSQC->',3F12.5,/,10X,I6,2X,8A1,2X,I6,/)
0171 DO 512 J1=1,40
0172 511 IF(IPEEK('17514').NE.'200') GOTO 511
0174 CALL IPOKER('17515,OUTQ(J1))
0175 512 CONTINUE
0176 CALL DELAY
0177 DGO=1
0178 IF(DFLAG.NE.0) GOTO 552
C
C***** BWSA *****
C
0180 915 IF(I.EQ.100) GOTO 2245
0182 IF(I.EQ.200) GOTO 2245
0184 IF(I.EQ.300) GOTO 2245
0186 GOTO 150
0187 2245 LINK=I/100
0188 IGRN=0
0189 IBX=1000+LINK
0190 2200 DO 2350 J=1,13
0191 JJ=J
0192 RN=RN(I1,I2)
0193 IF(RN.GE..998.OR.RN.LE..003) GOTO 2400
0195 IF(RN.GE..990.OR.RN.LE..005) GOTO 2450
0197 VMEAS(J)=RN*BRANGE(J,LINK)+ELL(J,LINK)
0198 IGRN=1

```

IGREEN VALS

PAGE 005

```

FORTRAN IV      V02.1-11

0199      GOTO 2500
0200      VMEAS(JJ)=RN*BRANGE(JJ, LINK)+BLL(JJ, LINK)
0201      PRED(IX, 1)=PRED(IX, 1)+1
0202      PRED(IX, 2)=JJ
0203      PRED(IX, 3)=7*MEAS(JJ)
0204      IF(JJ.EQ. 13) GOTO 2500
0205      DO 2550 K=JJ+1, 13
0206      VMEAS(K)=5*BRANGE(K, LINK)+BLL(K, LINK)
0207      GOTO 2500
0208      VMEAS(JJ)=RN*BRANGE(JJ, LINK)+BLL(JJ, LINK)
0209      GOTO 2600
0210      IS=JICVT(LINK, I14)
0211      CALL IMAGE(I14)
0212      RN=RN*(I1, I2)
0213      MON=RN*1000
0214

C
C      PWSQ REQUESTS A INTERRUPT IF FLAG SET
C
0215      IF(BFLAG.EQ. 0) GOTO 663
0216      IBX=1000+BFLAG
0217      DO 664 K=1, 13
0218      RN=RN*(I1, I2)
0219      VMEAS(K)=RN*BRANGE(K, BFLAG)+BLL(K, BFLAG)
0220      IF(PRED(IX, 1).EQ. 0) GOTO 665
0221      IS=JAFIX(PRED(IX, 2), K4)
0222      IS=IJCVT(K4, K2)
0223      K=K2
0224      VMEAS(K)=PRED(IX, 3)
0225      IS=JICVT(BFLAG, I14)
0226      CALL IMAGE(I14)
0227      RN=RN*(I1, I2)
0228      MON=RN*1000
0229      BGO=0
0230      GOTO 663
0231      CONTINUE
0232

C
0233      IF(BGO.EQ. 1) GOTO 150
0234      ICNT=66
0235      ENCODE(2, 601, OUTQ(1)) ICNT
0236      ENCODE(4, 501, OUTQ(3)) VMEAS(1)
0237      ENCODE(4, 501, OUTQ(7)) VMEAS(2)
0238      ENCODE(4, 501, OUTQ(11)) VMEAS(3)
0239      ENCODE(4, 501, OUTQ(15)) VMEAS(4)
0240      ENCODE(4, 501, OUTQ(19)) VMEAS(5)
0241      ENCODE(4, 501, OUTQ(23)) VMEAS(6)
0242      ENCODE(4, 501, OUTQ(27)) VMEAS(7)
0243      ENCODE(4, 501, OUTQ(31)) VMEAS(8)
0244      ENCODE(4, 501, OUTQ(35)) VMEAS(9)
0245      ENCODE(4, 501, OUTQ(39)) VMEAS(10)
0246      ENCODE(4, 501, OUTQ(43)) VMEAS(11)
0247      ENCODE(4, 501, OUTQ(47)) VMEAS(12)
0248      ENCODE(4, 501, OUTQ(51)) VMEAS(13)
0249      ENCODE(2, 515, OUTQ(55)) LINK
0250      ENCODE(8, 516, OUTQ(57)) BWD
0251      OUTQ(64)=ENC(4)
0252

```

AD-A078 391

BURROUGHS CORP PAOLI PA FEDERAL AND SPECIAL SYSTEMS GROUP F/G 9/2
SOFTWARE MAINTENANCE MANUAL FOR THE MODULAR SYSTEM CONTROL DEVE--ETC(U)
NOV 79 DCA100-76-C-0083

UNCLASSIFIED

66157

SBIE-AD-E100 313

NL

5 OF 5
AD
A078391



END
DATE
FILMED
1-80
DDC

PAGE 005

```

FORTRAN IV      V02.1-11
0253      ENCODE(2,517,OUTQ(65)) MON
0254      FORMAT(A2)
0255      516 FORMAT(A8)
0256      517 FORMAT(A2)
D 0257      WRITE(7,902)(VMEAS(IC),IC=1,13),LINK,(OUTQ(ID),ID=57,64),
D 0258      MON
D 0259      3F12.6,/,14X,16,2X,8A1,2X,16,/)
D 0260      DO 519 J1=1,80
D 0261      IF(IPEEK(175614)) .NE. "200" GOTO 518
D 0262      CALL IPOKEB(175616,OUTQ(J1))
D 0263      CONTINUE
D 0264      CALL DELAY
D 0265      BGO=1
D 0266      IF(BFLAG .NE. 0) GOTO 658
C
0267      IF(ICRN .EQ. 0) GOTO 150
0268      RN=RAN(11,12)
0269      IF(RN .LT. .9) GOTO 150
0270      RN=RAN(11,12)
0271      IEQ=3*RN*.5
0272      IF(IEQ .EQ. 0) IEQ=1
0273      ICNT=14
0274      ENCODE(2,601,OUTQ(1)) ICNT
0275      ENCODE(4,520,OUTQ(3)) ALARM
0276      ENCODE(2,521,OUTQ(11)) LINK
0277      ENCODE(2,522,OUTQ(13)) IEQ
0278      520 FORMAT(A4)
0279      521 FORMAT(A2)
0280      522 FORMAT(A2)
D 0281      DO 523 J1=1,80
D 0282      IF(IPEEK(175614)) .NE. "200" GOTO 524
D 0283      CALL IPOKEB(175616,OUTQ(J1))
D 0284      CONTINUE
D 0285      CALL DELAY
D 0286      WRITE(7,903)LINK,IEQ
D 0287      903 FORMAT(1X,'REWSA->', 'XXXX',2X,14,2X,14,/)
D 0288      C*****
C*****
C*****
0289      150 CONTINUE
0290      GOTO 100
0291      STOP
0292      END
C$$$

```

!NO ALARM

!EQUIP

!START OVER PROG

PAGE 021

```

FORTRAN IV      V02.1-11
0001      SUBROUTINE IMAGE(IN)
0002      BYTE ENC
0003      INTEGER*4 IN
0004      COMMON/EXP/ENC(4)
0005      ENCODE(4,1,ENC) IN
0006      FORMAT(4I)
0007      DO 2 J=1,4
0008      IF(ENC(J).LE.57 .AND. ENC(J).GE.48) GOTO 2
0009      ENC(J)='0'
0010      2 CONTINUE
0011      RETURN
0012      END
0013      C$$$$$$$

```

PAGE 001

```
FORTRAN IV      V02.1-11
0001      SUBROUTINE DELAY
0002      DO 1 J1=1,2
0003      DC 2 J2=1,10000
0004      2      CONTINUE
0005      1      CONTINUE
0006      RETURN
0007      END
```


J
SIG.MACRO

MACRO V03.02B 01:23:36 PAGE 1

```

1  .TITLE SIG.MACRO
2  .IDENT/V1.0/
3  .GLOBL INTVEC,REV5QC,RDD5QC,RDE5SA
4  .PSECT
5
6  ;INTERFACE ADDRESS
7
8
9  ;DSQC
10 RCSR1=177510
11 RBUF1=177512
12 XCSR1=177514
13 XBUF1=177516
14
15 ;VSQC
16 RCSR2=177412
17 RBUF2=177412
18 XCSR2=177414
19 XBUF2=177416
20
21 ;B5SA
22 RCSR3=175610
23 RBUF3=175612
24 XCSR3=175614
25 XBUF3=175616
26
27 ;VECTOR INIT
28
29 INTVEC: RESET
30     MOV     #100, R0
31     MOV     #TIME, (R0)
32     MOV     #320, R0
33     MOV     #RDV5QC, (R0)+
34     MOV     #340, (R0)+
35     BIS     #100, @RCSR2
36     MOV     #310, R0
37     MOV     #RDD5QC, (R0)+
38     MOV     #340, (R0)+
39     BIS     #100, @RCSR1
40     MOV     #300, R0
41     MOV     #RDE5SA, (R0)+
42     MOV     #340, (R0)+
43     BIS     #100, @RCSR3
44     MOV     #000, R0
45     MTPS    R0
46     RTS     PC
47     RTI
48
49     TIME:
50
51     RDV5QC: MOV     R0, -(SP)
52     TSTB     @RCSR2
53     BPL      1$
54     MOV     @RBUF2, R0
55     BIC     #177400, R0
56     CMPB    #122, R0
57     BNE      4$
58     MOV     #0, VGO
59     BR       1$

```

```

SIG.MACRO      MACRO V03.02E 01:22:36 PAGE 1-1

58 000150 010067 000002'      4$: MOV      VFLAG
59 000154 012600      1$: MOV      (SP)+, R0
60 000156 000002      RTI
61
62 000160 010046      RDSQC: MOV      R0, -(SP)
63 000162 105737      TSTB      G#RCSH1
64 000166 100015      BPL      2$
65 000170 113700      MOV      G#RBUF1, R0
66 000174 042700      BIC      #177400, R0
67 000200 122700      CMPB      #122, R0
68 000204 001004      BNE      5$
69 000206 012767      MOV      #0, LGO
70 000214 000402      BR      2$
71 000216 010067      5$: MOV      DFLAG
72 000222 012600      2$: MOV      R0, (SP)+, R0
73 000224 000002      RTI
74
75 000226 010046      EDW$A: MOV      R0, -(SP)
76 000230 105737      TSTB      G#RCSH3
77 000234 100015      BPL      3$
78 000236 113700      MOV      G#RBUF3, R0
79 000242 042700      BIC      #177400, R0
80 000246 122700      CMPB      #122, R0
81 000252 001004      BNE      6$
82 000254 012767      MOV      #0, BGO
83 000262 000402      BR      3$
84 000264 010067      6$: MOV      EFLAG
85 000270 012600      3$: MOV      R0, (SP)+, R0
86 000272 000002      RTI
87
88
89 000000
90
91 000000      .PSECT INT,RW,D,CBL,BLL,OVR
92 000000      DFLAG: .WORD 0
93 000000      VFLAG: .WORD 0
94 000000      BFLAG: .WORD 0
95 000000      LGO: .WORD 0
96 000000      VGO: .WORD 0
97 000000      BGO: .WORD 0
98
99 000000      .END INTVEC

```

SIG-MACRO MACRO V03.02B 01:22:36 PAGE 1-2

SYMBOL TABLE

BFLAG	000004R	002 RBUF1 = 177512	RCSE3 = 175512	VFLAG	000002R	002 XBUF3 = 175616
EGC	000012R	002 RBUF2 = 177412	RDE*CA	000226RG	002 ICSE1 = 177514	
DFLAG	000000R	002 RBUF3 = 175612	RDE*SC	000120RG	002 ICSE2 = 177414	
LGC	000000R	002 RCSE1 = 177510	RDE*SC	000112RG	002 ICSE3 = 175614	
INTVEC	000000RG	002 RCSE2 = 177412	TIME	000110R		
.ABS.	000000	000				
INT	000274	001				
INT	000214	002				
ERRORS DETECTED:	0					

VIRTUAL MEMORY USED: 294 WORDS (2 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 56 PAGES
 DE:SIGMAC,DK:SIGMAC=DR:SIGMAC.W20

FORTRAN IV V01C-03 FRI 03-JUN-77 12:24:53 PAGE 001
 CORE=08K, UIC=120,201 SDCAS.0BJ,SDCAS.LST=SDCAS.FOR

```

C THIS PROGRAM -SIM(SDCAS.FOR) WRITES SIMULATED SWITCH
C CONDITION REPORTS TO SDCA LOOP 5
C THIS PROGRAM RUN ON HOST 11/40
C
0001 DIMENSION ICRIT(2,16),IRPT(16)
0002 CALL ASSIGN(1,'TI:')
0003 DATA ICRIT(1,1),ICRIT(1,2),ICRIT(1,3)/1,513,25/
0004 DATA ICRIT(1,4),ICRIT(1,5),ICRIT(1,6)/25,10,50/
0005 DATA ICRIT(1,7),ICRIT(1,8),ICRIT(1,9)/50,10,10/
0006 DATA ICRIT(1,10),ICRIT(1,11),ICRIT(1,12)/10,128,128/
0007 DATA ICRIT(1,13),ICRIT(1,14),ICRIT(1,15)/128,128,10/
0008 DATA ICRIT(1,16)/10/
0009 DATA ICRIT(2,1),ICRIT(2,2),ICRIT(2,3)/2,256,10/
0010 DATA ICRIT(2,4),ICRIT(2,5),ICRIT(2,6)/10,25,40/
0011 DATA ICRIT(2,7),ICRIT(2,8),ICRIT(2,9)/40,5,5/
0012 DATA ICRIT(2,10),ICRIT(2,11),ICRIT(2,12)/10,64,64/
0013 DATA ICRIT(2,13),ICRIT(2,14),ICRIT(2,15)/64,64,5/
0014 DATA ICRIT(2,16)/5/
0015 I1=0
0016 I2=0
0017 I10=1
0018 I1=ID0
0019 R1=IRAN(I1,I2)
0020 DO 20 I=3,16
0021 ICRIT(1)=ICRIT(I10,I)*R1
0022 IF(R1 .LE. .98) GOTO 100
C ELSE SATURATED COND.
0024 IR2=13*IRAN(I1,I2)
0025 IF(IR2 .LT. 3) IR2=3
0027 IF(IR2 .GE. 16) IR2=16
0029 IRPT(IR2)=(R1*5)+ICRIT(I10,IR2)
0030 IRPT(2)=IRPT(11)+IRPT(12)+IRPT(13)+IRPT(14)
0031 WRITE(1,110)(IRPT(I),I=1,16)
0032 WRITE(2,110)(IRPT(I),I=1,16)
0033 FORMAT(1X,16I4)
C WAIT 5.5 SEC
0034 T1=SECND(0.)
0035 DELTA=SECND(T1)
0036 IF(DELTA .LT. 5.5) GOTO 150
0038 IF(ID0 .EQ. 2) GOTO 200
0040 I10=2
0041 GOTO 10
0042 STOP
0043 END

```


PAGE 001

12:00:00

01-JUL-79

COMP20.COM

PROGRAM:

COMPILER

FOR SIGGEN/NO LINE
MAC SIGMA

PAGE 001

12:00:00

01 JUL 79

1000.0000

PROGRAM:

GCMC1001

ASS D20 DK

R LTR

DX1:STGCN,DX1:MAC-DX1:STGCN/RE/CE/W.1

DX1:STGCN/77

451DK1

ASS DX1 DK